UNIVERSIDAD POLITÉCNICA DE MADRID

ESCUELA TÉCNICA SUPERIOR DE INGENIEROS AERONÁUTICOS

**Desarrollo, implementación y evaluación de un algoritmo
de lógica difusa para el control de actitud de
nanosatélites**

Tesis Doctoral

Autor: Daniel Jaime Calvo Cerezo

Ingeniero Aeronáutico

Madrid, Julio de 2017

DEPARTAMENTO DE AERONAVES Y VEHÍCULOS ESPACIALES

ESCUELA TÉCNICA SUPERIOR DE INGENIEROS AERONÁUTICOS

Tesis Doctoral

**Desarrollo, implementación y evaluación de un algoritmo
de lógica difusa para el control de actitud de
nanosatélites**

Autor:

Daniel Jaime Calvo Cerezo

Ingeniero Aeronáutico

Directoras de la Tesis Doctoral:

Ana Laverón Simavilla

María Victoria Lapuerta Gonzalez

Doctora Ingeniera Aeronáutica

Doctora Ingeniera Aeronáutica

Madrid, Julio 2017

Tribunal nombrado por el Sr. Rector Magfco. de la Universidad Politécnica de Madrid el día de de 2017.

Presidente:

Vocal:

Vocal:

Vocal:

Secretario:

Suplente:

Suplente:

Realizado el acto de defensa y lectura de la Tesis el día de de 2017 en la E.T.S.I. / Facultad

Calificación

EL PRESIDENTE

LOS VOCALES

EL SECRETARIO

A mi hija.

Índice

Índice	I
Lista de acrónimos	V
Lista de figuras	VII
Lista de tablas	XIII
Lista de símbolos	XV
Agradecimientos	XIX
Resumen	XXI
Abstract	XXIII
1. Introducción	- 1 -
1.1. Planteamiento del problema	- 1 -
1.2. Estado del arte	- 3 -
1.3. Objetivos	- 4 -
1.4. Estructura de la tesis.	- 5 -
2. El entorno espacial y las misiones espaciales	- 7 -
2.1. Clasificación de satélites	- 7 -
2.1.1. <i>Clasificación por el tipo de órbita</i>	- 7 -
2.1.2. <i>Clasificación por tamaño</i>	- 9 -
2.1.3. <i>Clasificación por el tipo de misión</i>	- 9 -
2.2. El entorno espacial y las misiones espaciales	- 10 -
2.2.1. <i>Efecto de las perturbaciones en la actitud del satélite</i>	- 13 -
3. Determinación y control de órbita y actitud en satélites	- 15 -
3.1. Representación de la actitud	- 15 -
3.1.1. <i>Ángulos de Euler.</i>	- 15 -
3.1.2. <i>Cuaterniones o parámetros de Euler-Rodrigues.</i>	- 17 -
3.1.3. <i>Matriz de giro o de cosenos directores.</i>	- 18 -
3.2. Control de actitud	- 18 -
3.3. Determinación de actitud	- 20 -
3.4. Sensores	- 21 -
3.5. Actuadores	- 22 -

3.6. Métodos de estabilización pasivos	- 23 -
4. Algoritmos de control	- 25 -
4.1. Introducción	- 25 -
4.2. PID	- 25 -
4.3. Lógica difusa	- 28 -
4.3.1. Control basado en lógica difusa	- 31 -
4.3.1.1. Proceso de Fuzzificación	- 31 -
4.3.1.2. Reglas	- 31 -
4.3.1.3. Proceso de Defuzzificación	- 32 -
5. El proyecto QB50 y el satélite QBITO	- 35 -
5.1. Introducción	- 35 -
5.2. El proyecto QB50	- 35 -
5.3. El satélite QBITO	- 36 -
5.3.1. La misión del satélite QBITO	- 37 -
5.3.2. Subsistemas de QBITO	- 39 -
5.3.3. Subsistema de potencia	- 41 -
5.3.4. Subsistema del ordenador de a bordo	- 41 -
5.3.5. Módulo de comunicaciones	- 42 -
5.3.6. Módulo de control de actitud	- 43 -
5.3.7. Actuadores de QBITO	- 46 -
6. Elaboración de las herramientas de soporte necesarias	- 47 -
6.1. Sensores y actuadores	- 47 -
6.2. Simulador de actitud	- 50 -
6.2.1. Perturbaciones ambientales	- 53 -
7. Diseño de los controladores	- 57 -
7.1. PID	- 57 -
7.2. Lógica difusa	- 59 -
7.2.1. Desaturación de la rueda de momento	- 66 -
7.3. Calibración	- 68 -
8. Análisis y comparación	- 73 -
8.1. Verificación y análisis.	- 73 -

8.1.1.	<i>Test 1</i>	- 73 -
8.1.2.	<i>Test 2</i>	- 74 -
8.1.3.	<i>Test 3</i>	- 75 -
8.2.	Control Difuso-Adaptativo	- 81 -
8.3.	Calibración a baja altura	- 83 -
8.3.1.	<i>Test 3.2</i>	- 85 -
8.3.2.	<i>Test 4</i>	- 87 -
8.3.3.	<i>Test 5</i>	- 90 -
8.3.4.	<i>Conclusiones</i>	- 92 -
9.	Análisis multivariable tipo Monte Carlo	- 93 -
9.1.1.	<i>Escenario</i>	- 94 -
9.1.2.	<i>Análisis de datos</i>	- 98 -
9.1.3.	<i>Conclusiones</i>	- 107 -
10.	<i>Software</i> embarcado	- 109 -
10.1.	Diseño conceptual	- 109 -
10.1.1.	<i>Funciones para el control de actitud: código Matlab autogenerado</i>	- 118 -
10.1.2.	<i>Propagador orbital</i>	- 120 -
10.1.3.	<i>Cálculo de efemérides</i>	- 121 -
10.1.4.	<i>Determinación de actitud</i>	- 122 -
10.1.5.	<i>Filtro de Kalman</i>	- 123 -
10.1.6.	<i>Control de actitud</i>	- 124 -
10.2.	Implementación del SW embarcado	- 125 -
10.2.1.	<i>Ensayos en placa de prototipado</i>	- 125 -
10.2.2.	<i>Implementación en el CubeComputer</i>	- 125 -
11.	QBITO en órbita	- 127 -
11.1.	Introducción	- 127 -
11.2.	Proceso previo	- 127 -
11.2.1.	<i>Fabricación e Integración</i>	- 127 -
11.2.2.	<i>Lanzamiento y despliegue</i>	- 131 -
11.3.	Operaciones en órbita y análisis de resultados	- 132 -

12. Conclusiones	- 135 -
13. Bibliografía	- 139 -
Anexo A: sistemas de referencia	- 145 -
A.1 Ejes inerciales o ECI	- 145 -
A.2 Ejes fijos a la tierra o ECEF	- 146 -
A.3 Ejes órbita u ORF	- 146 -
A.4 Ejes cuerpo o BRF	- 147 -

Lista de acrónimos

ADCS	<i>Attitude Determination and Control Subsystem.</i>
ARM	<i>Advanced RISC Machine.</i>
bps	<i>Bits Per Second.</i>
BRF	<i>Body Reference Frame.</i>
COM	Subsistema de comunicaciones.
DA	Difuso-Adaptativo.
DKE	<i>Dynamics, Kinematics and Environment.</i>
DVK	<i>Development Kit.</i>
ECEF	<i>Earth Centered, Earth Fixed.</i>
ECI	<i>Earth Centered Inertial.</i>
EPS	<i>Electric Power Subsystem.</i>
ESW	<i>Experimental Software.</i>
E-USOC	<i>Spanish User Support and Operations Centre.</i>
FOV	<i>Field Of View.</i>
GEO	<i>Geosynchronous Earth Orbit.</i>
GMT	<i>Greenwich Mean Time.</i>
HEO	<i>High Earth Orbit.</i>
I2C	<i>Inter-Integrated Circuit.</i>
IGRF	<i>International Geomagnetic Reference Field.</i>
INMS	<i>Ion Neutral Mass Spectrometer.</i>
ISS	<i>International Space Station.</i>
JAXA	<i>Japanese Aerospace Exploration Agency.</i>
LEO	<i>Low Earth Orbit.</i>
LQR	<i>Linear Quadratic Regulator.</i>

MEO	<i>Medium Earth Orbit.</i>
MWIRD	<i>Medium Wave InfraRed Detector.</i>
NASA	<i>National Aeronautics and Space Agency.</i>
NORAD	<i>North American Aerospace Defense Command.</i>
NRLMSISE	<i>Naval Research Laboratory Mass Spectrometer and Incoherent Scatter Radar until Exosphere.</i>
OBC	<i>Onboard Computer.</i>
ORF	<i>Orbit Reference Frame.</i>
PCB	<i>Printed Circuit Board.</i>
PCM	<i>Phase Changing Material.</i>
PID	Proporcional, Integral y Derivativo.
RAM	<i>Random Access Memory.</i>
SD	<i>Secure Digital.</i>
SEU	<i>Single Event Upset.</i>
SGP4	<i>Simplified General Perturbations model 4.</i>
SRAM	<i>Static Random Access Memory.</i>
TLE	<i>Two Line Elements.</i>
UART	<i>Universal Asynchronous Receiver-Transmitter.</i>
UML	<i>Unified Modelling Language.</i>
UHF	<i>Ultra hHigh Frequency.</i>
VHF	<i>Very High Frequency.</i>

Lista de figuras

Figura 2.1. Esquema de clasificación de órbitas según su altura.....	- 8 -
Figura 2.2. Representación, no a escala, del problema de la basura espacial, principalmente en órbitas LEO y GEO donde se observa una preocupante acumulación de elementos. Crédito: Agencia Espacial Europea.....	- 10 -
Figura 2.3. Pares de perturbación esperables para un CubeSat según la altura de la órbita para las órbitas típicas.....	- 14 -
Figura 3.1. Representación de la actitud del sistema $X_B Y_B Z_B$ respecto al $X_0 Y_0 Z_0$. <i>Roll</i> , <i>pitch</i> y <i>yaw</i> son los nombres en inglés para los tres ángulos de Euler: balanceo, cabeceo y guiñada; v y δ son el vector y el ángulo respectivamente que representarían el giro mediante cuaterniones (sección 3.1.2).	- 16 -
Figura 3.2. Esquema general de control. “Comp” es la abreviatura de comparador.	- 19 -
Figura 4.1. Esquema de un control de tipo PID.	- 26 -
Figura 4.2. Cambio en la respuesta de un control PID al verse modificadas sus ganancias. La gráfica superior izquierda muestra el efecto de modificar K_p . La gráfica superior derecha muestra el efecto de modificar K_d . Por último, la gráfica inferior muestra el efecto de modificar la ganancia K_i . Fuente: Wikipedia.....	- 27 -
Figura 4.3. Clasificación de la temperatura del agua mediante lógica booleana.	- 29 -
Figura 4.4. Ejemplo de funciones de pertenencia. Se ha llamado a la etiqueta “Templada” como “OK” por razones estéticas. Como referencia se ha calculado la pertenencia de una temperatura de 22.5 grados (0.75 “Fría” y 0.25 “Templada”).	- 30 -
Figura 4.5. Esquema de funcionamiento de un control basado en lógica difusa.	- 31 -
Figura 4.6. Proceso de <i>defuzzificación</i> usando el método Mamdani: la actuación viene marcada por el centro de áreas de la unión de funciones de pertenencia marcadas por las reglas, recortadas según el grado de pertenencia de las entradas correspondiente.	- 33 -
Figura 5.1. El satélite QBITO junto al ingeniero de actitud Daniel Calvo.	- 37 -
Figura 5.2. Vista abierta del satélite QBITO con los experimentos y actuadores (sección 5.3.7) etiquetados.....	- 38 -

Figura 5.3. Requisitos impuestos por el INMS para la misión QBITO. Los ejes X_O , Y_O y Z_O representan el sistema de referencia ORF y los X_B , Y_B y Z_B son los ejes BRF.	- 39 -
Figura 5.4. Esquema de conexiones de QBITO (Barrios, 2013).	- 40 -
Figura 5.5. Módulo de baterías y EPS.	- 41 -
Figura 5.6. Placa de Cubecomputer. Tanto el OBC como el ADCS usan una placa como esta como cerebro del subsistema.	- 42 -
Figura 5.7. <i>Hardware</i> del módulo del ADCS. Control, Computer y Sense son las placas del ADCS: CubeControl, CubeComputer y CubeSense respectivamente.	- 44 -
Figura 5.8. El ingeniero de sistemas de QBITO sostiene el módulo del ADCS completo.	- 45 -
Figura 6.1. Esquema simplificado del control.	- 48 -
Figura 6.2. Diferencias en actitud tras actuación a dipolo constante en el eje Y_B (0.15 Am^2) usando el modelo original con respecto al uso del nuevo modelo (actuación de 0.2 Am^2 durante el 75% del tiempo).	- 50 -
Figura 6.3. Orden de magnitud de los pares de perturbación en función de la altura de la órbita.	- 54 -
Figura 7.1. Esquema del controlador difuso. Los bloques llamados G son las ganancias que hay que calibrar.	- 60 -
Figura 7.2. Reglas de pertenencia difusas para el ángulo σ	- 62 -
Figura 7.3. Reglas de pertenencia difusas para los ángulos γ y μ	- 63 -
Figura 7.4. Detalle de la zona central de la figura 7.3.	- 63 -
Figura 7.5. Funciones de pertenencia difusas para la velocidad angular orbital alrededor del eje Y_B (Ω_{OBy}).	- 64 -
Figura 7.6. Funciones de pertenencia difusas para las velocidades angulares orbitales en los ejes X_B y Z_B (Ω_{OBx} y Ω_{OBz}).	- 64 -
Figura 7.7. Representación del controlador difuso usado en el eje Y_B	- 66 -
Figura 7.8. Funciones de pertenencia para la desaturación de la rueda.	- 67 -
Figura 8.1. Representación del equivalente del error tridimensional (ec. (8.1)) entre dos vectores en lugar de cuaterniones.	- 74 -
Figura 8.2. Error medio a lo largo de una órbita en relación a la altura de la órbita.	- 75 -
Figura 8.3. Respuestas superpuestas de ambos controles a excitación en cada uno de los ángulos por separado. Cada línea representa una simulación diferente, y todas ellas se han hecho a 380 km de altura.	- 76 -
Figura 8.4. Respuestas superpuestas de ambos controladores a baja altura (200 km).	- 77 -

Figura 8.5. Aceleraciones sufridas por el satélite en las maniobras de balanceo y guiñada con ambos controladores.....	- 78 -
Figura 8.6. Error residual tras una maniobra de cincuenta grados en distintas alturas a lo largo de toda la misión. La línea negra horizontal punteada representa el límite impuesto en la calibración.....	- 79 -
Figura 8.7. Respuesta del PID al comando en guiñada. Se observa como el ángulo de balanceo adquiere un nuevo punto de equilibrio distinto del valor cero.....	- 80 -
Figura 8.8. Coste del control difuso escalado usando el coste del PID.	- 81 -
Figura 8.9. Esquema del control DA. T_{dj} es la perturbación estimada en el eje j	- 82 -
Figura 8.10. Funciones de pertenencia del control DA: a la izquierda las funciones de pertenencia en la variable de entrada y a la derecha para la variable de salida.....	- 83 -
Figura 8.11. Ratio de costes para maniobras de cincuenta grados en cada uno de los ejes, entre el control difuso original y dos controles de tipo PID, uno optimizado para baja altura y otro para alta (calibración original).	- 84 -
Figura 8.12. Error residual del control DA en cabeceo comparado con el original, calibrado para órbitas altas y para órbitas bajas.....	- 85 -
Figura 8.13. Error residual del control DA en guiñada comparado con el original, calibrado para órbitas altas y para órbitas bajas.....	- 86 -
Figura 8.14. Ratio de costes del control DA en cabeceo comparado con el original, calibrado para órbitas altas y para órbitas bajas.....	- 86 -
Figura 8.15. Ratio de costes del control DA en guiñada comparado con el original, calibrado para órbitas altas y para órbitas bajas.....	- 87 -
Figura 8.16. Tiempo requerido por ambos algoritmos de control para alcanzar una actitud nominal estable tras empezar con condiciones iniciales dinámicas en cada uno de los ejes principales del satélite.	- 89 -
Figura 8.17. Respuesta en frecuencia para una excitación sinusoidal de veinte grados de amplitud.....	- 91 -
Figura 9.1. Esquema del método de Monte Carlo. t es el tiempo simulado y N es el número de repeticiones y puede ser tan grande como se necesite.	- 93 -
Figura 9.2. Distribución de probabilidad de los nuevos momentos de inercia.	- 95 -
Figura 9.3. Distribución espacial de los puntos iniciales para las simulaciones.	- 96 -

Figura 9.4. Matriz de correlaciones entre variables de entrada y entre variables de entrada y variables de salida de las simulaciones de Monte Carlo.	- 99 -
Figura 9.5. Error tridimensional instantáneo. Cada línea representa una simulación independiente con parámetros aleatorios diferentes.	- 100 -
Figura 9.6. Tiempo necesario hasta estar "cerca" de la solución (ec. (9.6)).	- 101 -
Figura 9.7. Histograma del orden de magnitud del error residual.....	- 102 -
Figura 9.8. Momentos de inercia descartados. Las líneas azules verticales representan el umbral seleccionado para cada momento de inercia.	- 103 -
Figura 9.9. Error tridimensional instantáneo de todas las simulaciones salvo las correspondientes a los momentos de inercia degenerados.....	- 104 -
Figura 9.10. Histograma del orden de magnitud del error residual para las simulaciones no degeneradas.	- 104 -
Figura 9.11. Dispersión de los logaritmos de los cocientes de errores y costes entre ambos controles. Las líneas punteadas marcan los valores medios.	- 105 -
Figura 9.12. Dispersión de los logaritmos de los ratios de errores y costes con las líneas de nivel de densidad superpuestas. Entre dos curvas consecutivas se encierran aproximadamente el 10% de los puntos.	- 107 -
Figura 10.1. Diagrama de secuencia de funcionamiento del ADCS de QBITO.....	- 111 -
Figura 10.2. Detalle del proceso de cambio de SW o arranque del ADCS.....	- 112 -
Figura 10.3. <i>Bootloader</i> del ADCS.	- 113 -
Figura 10.4. Esquema de secuencia de funcionamiento del ESW.	- 115 -
Figura 10.5. Esquema de "clases" de las funciones autogeneradas. Las flechas representan las dependencias entre bloques.	- 119 -
Figura 11.1. QBITO sin los paneles laterales.....	- 128 -
Figura 11.2. QBITO ya finalizado, a falta de integrar con el INMS.....	- 129 -
Figura 11.3. INMS integrado con QBITO.....	- 130 -
Figura 11.4. Despegue del lanzador Atlas V con la nave Cygnus en su interior el día 18 de Abril de 2017. Crédito: <i>United Launch Alliance</i>	- 130 -
Figura 11.5. Despliegue de QBITO (abajo) desde el dispensador de Nanoracks (arriba).	- 131 -

Figura A.1. Representación de los ejes ECI y ECEF durante el equinoccio de primavera.....	- 145 -
Figura A.2. Representación de los ejes ECEF, ORF y BRF.	- 147 -

Lista de tablas

Tabla 7.1. Tabla de abreviaturas que aparecen en las funciones de pertenencia y en la tabla de reglas del control difuso.....	- 62 -
Tabla 7.2. Reglas difusas para el eje Y_B . Los valores numéricos representan la fracción de actuación con respecto al valor máximo.....	- 65 -
Tabla 7.3. Reglas difusas para los ejes X_B y Z_B . Los valores numéricos representan la fracción de actuación con respecto al valor máximo.....	- 65 -
Tabla 7.4. Reglas difusas para el control de desaturación de la rueda. En cada celda se representan los valores de actuación como fracción del par máximo.	- 68 -
Tabla 7.5. Reglas difusas para el control de desaturación de la rueda. En cada celda se representan los que toma la variable S	- 68 -
Tabla 7.6. Secuencia de excitación y simulación del proceso automatizado de calibración.	- 69 -
Tabla 7.7. Valores obtenidos para las ganancias de ambos controles mediante el proceso automatizado de calibración.....	- 71 -
Tabla 8.1. Ganancias del PID y del control difuso calibrados a baja altura.....	- 84 -
Tabla 10.1. Definición de las secuencias correspondientes a los test científicos.	- 116 -
Tabla 10.2. Modos de operación del ADCS de QBITO.....	- 118 -

Lista de símbolos

A	Area.
$a_1 a_2$	Ángulos genéricos.
B	Campo magnético terrestre.
c	Velocidad de la luz en el vacío.
$C(t)$	Función de costes.
C	Coste acumulado.
$\tilde{C}$	Ratio de costes.
C_D	Coefficiente adimensional de resistencia aerodinámica.
C_{DA}	Función de coste del control Difuso-Adaptativo.
C_{PID}	Funcion de coste del control PID.
d	Brazo usado para el cálculo de momentos.
D	Dipolo del satélite.
e	Error.
$\bar{e}$	Error medio.
$\tilde{e}$	Ratio de errores.
e_{3D}	Error tridimensional.
e_{DA}^*	Error del control Difuso-Adaptativo usado en el test Monte Carlo.
e_{MC}	Error residual máximo.
e_{PID}^*	Error del control PID usado en el test Monte Carlo.
F_s	Radiación solar.
H	Altura de la órbita.
$\vec{h}$	Vector momento angular orbital unitario.
I	Matriz de inercia del satélite.

I_{lat}	Máximo producto de inercia del satélite.
$I_X I_Y I_Z$	Momentos principales de inercia.
$\mathbf{I}_w$	Matriz de inercia de la rueda de momento.
J_2	Principal armónico esférico del desarrollo en serie del potencial terrestre.
K_{jp}	Ganancia proporcional del PID en el eje "j".
K_{jd}	Ganancia derivativa del PID en el eje "j".
K_{ji}	Ganancia integral del PID en el eje "j".
K_T	Constante física usada para usar la actuación magnética de la rueda en la función de costes.
K_w	Constante física usada para usar la aceleración de la rueda en la función de costes.
M	Momento magnético terrestre en los polos.
N	Numero de iteraciones en los test de Monte Carlo.
$\mathbf{q}$	Cuaternión genérico.
$\mathbf{q}'$	Cuaternión genérico no normalizado.
$\vec{\mathbf{q}}$	Parte vectorial del cuaternión.
q_0	Parte escalar del cuaternión.
$\mathbf{q}_{BO}$	Cuaternión de la actitud del satélite respecto al sistema ORF.
$\mathbf{q}_E$	Cuaternión del error.
$\mathbf{q}_{IB}$	Cuaternión de la actitud del satélite respecto al sistema ECI.
$\mathbf{q}_{IO}$	Cuaternión de la actitud del sistema ORF respecto al sistema ECI.
$\mathbf{q}_S$	Cuaternión del satélite.
$\mathbf{q}_T$	Cuaternión de la actitud objetivo.
$\hat{\mathbf{r}}$	Vector radio orbital unitario.

r	Reflexividad del satélite.
$\mathbf{R}$	Matriz de rotación.
R_T	Radio terrestre.
$\mathbf{T}$	Par vectorial.
T	Componente del par vectorial.
$\tilde{T}$	Par de control efectivo.
t	Tiempo de simulación.
T_{aero}	Par de perturbación por resistencia aerodinámica.
T_{cj}	Par de control en el eje "j".
T_{gg}	Par de perturbación por gradiente de gravedad.
T_{mag}	Par de perturbación por dipolo residual.
T_{max}	Máximo par dado por un actuador dado.
T_{sol}	Par de perturbación por presión solar.
$\mathbf{V}$	Vector velocidad orbital.
$\mathbf{X}_B\mathbf{Y}_B\mathbf{Z}_B$	Ejes fijos al cuerpo (BRF).
$\mathbf{X}_O\mathbf{Y}_O\mathbf{Z}_O$	Ejes fijos a la órbita (ORF).
α	Precisión en apuntamiento.
β	Precisión en determinación.
$\gamma \ \sigma \ \mu$	Ángulos equivalentes.
δ	Ángulo de la representación de una rotación mediante eje-ángulo.
Δe^*	Ratio de variación del error en las simulaciones Monte Carlo.
$\Delta\omega_w$	Aceleración angular de la rueda de momento.
ϑ	Cada uno de los 3 ángulos comandados (balanceo, cabeceo y guiñada).
θ	Segundo ángulo de Euler.

θ	Error inicial en el primer ángulo de Euler para los test de Monte Carlo.
μ	Constante de gravitación de la Tierra.
$\mathbf{v}$	Eje de giro en la representación de una rotación mediante eje-ángulo.
ρ	Densidad atmosférica.
σ_e	Desviación típica del error.
φ	Primer ángulo de Euler.
Φ	Error inicial en el primer ángulo de Euler para los test de Monte Carlo.
ψ	Tercer ángulo de Euler.
ψ	Error inicial en el primer ángulo de Euler para los test de Monte Carlo.
Ω	Velocidad angular del satélite respecto a ejes ECI.
Ω_{OB}	Velocidad angular respecto a ORF.
ω_w	Velocidad de rotación de la rueda de momento.

Agradecimientos

En primer lugar me gustaría agradecer a mis Directoras de Tesis, Ana Laverón Simavilla y Maria Victoria Lapuerta González su apoyo y sobre todo la oportunidad que me han brindado. También quiero mencionar a mis compañeros del E-USOC y a todos los miembros del proyecto QBITO que han logrado convertir en una realidad el satélite QBITO.

Una especial mención merecen tanto Taisir Avilés, pues su trabajo fundamentó las bases sobre las que construir, como todos los estudiantes que han colaborado con la misión.

El proyecto QBITO ha sido financiado por el E-USOC y por el proyecto europeo QB50.

Por último, pero no por ello menos importante, a mis padres, pues sin su apoyo a lo largo de mi vida no habría llegado tan lejos, y a mi mujer, por ser capaz de aguantarme casi siempre.

Resumen

El entorno de las misiones espaciales suele tender hacia el conservadurismo a la hora de implementar nuevas tecnologías en aras de ganar una mayor fiabilidad. Este hecho dificulta que se beneficie de nuevas tecnologías que podrían mejorar su desempeño. En concreto, la mayoría de misiones confían en controles clásicos para el control de actitud ya que éstos han sido extensivamente probados en órbita. El objetivo de esta tesis es implementar el uso del control inteligente en el campo de las misiones espaciales. En la tesis se diseña un control difuso adaptativo para el subsistema de control de actitud (ADCS) de un nanosatélite y se compara su eficiencia y desempeño con un control tradicional de tipo Proporcional, Integral y Derivativo (PID).

El control inteligente, basado en lógica difusa, se ha diseñado e implementado para ser probado en órbita a bordo de un nanosatélite (llamado QBITO), recientemente desplegado desde la Estación Espacial Internacional como parte de la misión QB50, cuyo objetivo principal es poner en órbita una constelación de nanosatélites que estudiará la composición de la termosfera. Un control de actitud eficiente es fundamental para completar los objetivos de la misión del satélite QBITO dado que la energía disponible está fuertemente limitada.

En esta tesis se ha analizado principalmente el consumo energético y la precisión de ambos controladores, difuso y PID, encontrándose que el controlador difuso es mucho más eficiente (hasta un 70% menos de energía requerida), consiguiendo una precisión mejor en términos generales que el PID. Analizando estadísticamente ambos algoritmos de control en un amplio rango de escenarios, se ha observado que, en promedio, el PID es hasta un 50% menos eficiente y un 250% menos preciso.

Finalmente, para acabar de validar el modelo de control desarrollado en esta tesis, y con él la conveniencia del uso de este tipo de algoritmos, éste va a ser probado en órbita a bordo del nanosatélite QBITO en los próximos meses.

En resumen, esta tesis muestra como el uso de este tipo de algoritmos inteligentes de control supone una gran ventaja frente a algoritmos clásicos que son los que se implementan habitualmente.

Abstract

The space missions tend towards being conservative when incorporating new technologies for the sake of reliability. That prevents benefiting from new technologies that may significantly improve their performances. Specifically, most missions rely on classic controllers for attitude control as they are thoroughly tested in orbit. The objective of this thesis is to implement the use of intelligent systems in the space missions' field. An adaptive fuzzy logic controller is designed for a nanosatellite attitude determination and control subsystem (ADCS) and its performance and efficiency are compared with a traditional Proportional, Integrative and Derivative (PID) controller.

The intelligent controller has been designed and implemented in order to be tested on board a nanosatellite (named QBITO), recently deployed on orbit from the International Space Station as a part of the QB50 mission, whose objective is to deploy a constellation of nanosatellites in orbit that will study the thermosphere composition. An efficient ADCS is essential to fulfill the QBITO satellite mission objectives as the available power is strongly constrained.

In this thesis have been analyzed mainly the power consumption and the accuracy of both controllers, fuzzy and PID, finding that the fuzzy controller is much more efficient (up to 70% less power required), achieving better precision in general than the PID. Statistically analyzing both controllers in a wider set of scenarios it is observed that, in average, the PID is up to 50% less efficient and 250% less accurate.

Finally, to complete the control model validation developed in this thesis and, at the same time, the suitability of this kind of algorithms, it will be tested in orbit during the following months onboard the QBITO satellite.

Summing up, this study shows that the use of this type of intelligent control systems represents a great advantage over conventional control systems currently being used in satellite attitude control.

1. INTRODUCCIÓN

En esta tesis se ha abordado el estudio del uso de controles basados en lógica difusa para el control de actitud de nanosatélites y se ha comparado su eficiencia y precisión con los de un método de control clásico. A continuación se presenta el estado del arte, los objetivos y la estructura de la tesis.

1.1. Planteamiento del problema

La actitud de un satélite es su orientación tridimensional respecto a un sistema de ejes determinado y, en general, viene impuesta por los requisitos de la misión. El subsistema de determinación y control de actitud es crítico para la mayoría de misiones espaciales. La inmensa mayoría de éstas tienen requisitos de actitud impuestos por sus cargas de pago para poder ser operadas satisfactoriamente. Estos requisitos impuestos pueden variar entre algunos grados (o incluso decenas de grados) en misiones de bajo presupuesto hasta milésimas de segundo de arco para algunas misiones científicas de mayor envergadura (Beals, 1988).

Además del requisito de precisión estático (i. e. la precisión en apuntamiento), la estabilidad de la actitud así como la robustez del control utilizado suelen ser también requisitos, cuando no explícitos, implícitos en la mayoría de misiones. Un control de actitud inestable puede acabar en un fracaso de la misión, o cuanto menos, afectar negativamente a su retorno científico (Harland, 2005) (JAXA, 2016). Por otro lado, un control robusto garantiza el cumplimiento de los requisitos a pesar de las perturbaciones externas o internas que puedan surgir de manera inesperada. Dado que el control debe actuar de forma inmediata y a lo largo de toda la misión del satélite, éste debe ser ejecutado a bordo del satélite, con todas las limitaciones propias de los sistemas embarcados.

Finalmente, el algoritmo de control debe ser lo más eficiente posible dado que la energía disponible para las actuaciones suele estar limitada, es decir: un control eficiente maximizará el retorno de la misión. La energía que utiliza el subsistema de actitud puede ser generada a bordo (i. e. potencia eléctrica) (Olson, 1993), puede ser obtenida mediante el uso de combustible (i. e. propulsión) o mediante una configuración mixta en la que se genera potencia eléctrica a bordo pero empleado combustible (Wolfgram, 1980).

En el entorno de las operaciones espaciales se han empleado o estudiado múltiples tipos de controladores. Sin embargo, a pesar de que el uso de sistemas de control basados en lógica difusa está muy extendido en numerosas áreas tecnológicas, ha sido usada en contadas misiones espaciales. La principal dificultad a la hora de implementar nuevos sistemas de control en el campo de las operaciones espaciales, es que los cambios han de ser introducidos siempre de manera lenta y cuidadosa dado que, como se ha indicado antes, un control defectuoso puede condenar toda una misión al fracaso. Por ello es necesario no solo un buen diseño y una simulación intensiva de los algoritmos de control, sino también la prueba y verificación de su buen funcionamiento en condiciones reales (en órbita).

Por todo ello es necesario comparar la eficiencia y precisión de los controladores difusos con los de los controladores clásicos que se emplean en las misiones espaciales. Sin embargo, no se ha encontrado en la literatura comparaciones de controladores difusos y controles de tipo proporcional, integral y derivativo (PID) para control de actitud de CubeSats (CubeSat, [online]), siendo los controladores PID los más utilizados y probados en el subsistema de control de actitud (ADCS por sus siglas en inglés: *Attitude Determination and Control Subsystem*) de satélites. Es por ello que la presente tesis se dedica al diseño y optimización de un controlador difuso adaptado para una misión de un nanosatélite, así como a la comparación de su rendimiento y eficiencia con un controlador PID tradicional para la misma misión.

Además, como parte de la tesis se desarrollará un *software* (SW) experimental que será probado en órbita en un futuro próximo como parte de la misión QB50. Esta misión es el principal objetivo del proyecto europeo QB50, un proyecto del 7º Programa Marco, cuyo objetivo es poner en órbita una constelación de 50 CubeSats, dedicada a la adquisición de medidas *in situ* y multipunto de los parámetros y constituyentes claves de la termosfera o ionosfera baja, y a la investigación de la re-entrada en la atmósfera. Toda la información sobre este proyecto puede encontrarse en (Von Karman Institute, [online]). Una propuesta, liderada por la profesora Ana Laverón Simavilla, fue seleccionada y se está llevando a cabo desde el año 2012. El satélite propuesto para esta misión se llama QBITO (Barrios, 2014).

1.2. Estado del arte

A lo largo de la historia, los algoritmos de control usados en los satélites han variado desde controles tradicionales (Wingrove, 1963), que se basan en teorías clásicas de control como el PID (Sidi, 2000) o Reguladores Lineales Cuadráticos (LQR por sus siglas en inglés: *Linear Quadratic Regulator*) (Souza, 2006), hasta controles inteligentes basados en lógica difusa (Ortega, 1995) (Steyn, 1994) (Nagi, 2010) (Guana, 2005) (Walker, 2015) o algoritmos con aprendizaje como las redes neuronales (Zou, 2010) (Fazlyab, 2016).

Analizando en concreto el empleo de la lógica difusa en algoritmos de control, por ser ésta el centro de esta tesis, se observa que es una teoría ya madura (Zadeh, 1965) (Lee, 1990) (Jantzen, 2013) (Takagi, 1985) que ha sido usada en algoritmos de control en numerosas aplicaciones comerciales (Østergaard, 1976) (Oshima, 1988) (Toshikazu, 1991) y que se caracteriza por simplificar el proceso de diseño, aprovechando el conocimiento ya adquirido de expertos seleccionados (Østergaard, 1976) (Oshima, 1988) (Layne, 1993). Los ingenieros que trabajan en el proceso de diseño de satélites tienen, habitualmente, una amplia experiencia en teorías de control así como amplias nociones en la dinámica de los satélites y el entorno espacial en general, que pueden ser aprovechadas para beneficio de la misión. Al usar el conocimiento ya adquirido, se puede reducir la duración de la etapa de diseño (con todos sus costes aparejados), muy útil en el caso de los pequeños satélites que suelen ser de bajo coste. Además, en el caso de tratarse de pequeños satélites (nano y pico) la potencia disponible suele estar fuertemente limitada lo que implica el uso de algoritmos eficientes. Estos dos factores hacen ideal el uso de controladores basados en lógica difusa para el control de actitud en nanosatélites. Sin embargo, a pesar de todas sus ventajas, la lógica difusa ha sido implementada en pocas misiones espaciales, aunque sí ha sido estudiada y los resultados obtenidos han sido muy satisfactorios (Ortega, 1995) (Steyn, 1994) (Nagi, 2010) (Guana, 2005) (Walker, 2015).

En trabajos anteriores se ha comparado el funcionamiento de un control tradicional, un LQR con un algoritmo de lógica difusa en un CubeSat (CubeSat, [online]) a través de simulaciones (Walker, 2015), obteniéndose que el control difuso implica un coste menor que el LQR, siendo además más rápido en llegar a la solución. En este trabajo se ha diseñado un control de actitud adaptable basado en lógica difusa para la misión de un nanosatélite. Además se ha estudiado el funcionamiento de este algoritmo comparándolo en tierra con un algoritmo tradicional de tipo PID que es el más usado, tanto

en satélites como en la mayoría de aplicaciones industriales. Esta comparación no ha sido llevada a cabo anteriormente y es por ello que se considera de gran interés. Por último, se espera poder probar ambos tipos de control en órbita, embarcados a bordo del satélite QBITO usando el mismo *hardware* en las mismas condiciones externas. Esta es, por lo tanto, una oportunidad única de probar *in situ* un control clásico con uno basado en inteligencia artificial.

1.3. Objetivos

Como ya se ha visto, existen prácticamente tantos algoritmos de control como problemas en los que utilizarlos. En la gran mayoría de casos éstos se pueden probar en el proceso que se quiere controlar (también llamado planta), no siendo así cuando la planta a controlar es la actitud de un satélite en órbita dado que es imposible conseguir reproducir las condiciones de ingravidez en las que la planta deberá funcionar. Por lo tanto, la oportunidad de comparar *in situ* dos algoritmos de control de actitud en la misma planta es única. El satélite QBITO, proporciona la plataforma ideal para ese ensayo comparativo. El desarrollo del satélite ha permitido generar desde cero un control de actitud completo y que éste se pruebe en órbita. Además, QBITO lleva embarcado también un control clásico ya validado, que garantice la consecución del resto de objetivos de la misión, lo que permite que se desarrollen algoritmos más experimentales que los usados tradicionalmente.

En esta tesis se va a estudiar el uso de algoritmos de lógica difusa adaptativos en comparación con un algoritmo clásico PID, que es el más usado en satélites. Además, se espera poder comparar en órbita usando el mismo satélite, el QBITO, el control clásico con el nuevo control difuso. Es, por lo tanto, una oportunidad única de comparar dos controles independientes, uno clásico y uno inteligente, en el mismo *hardware* con las mismas condiciones ambientales. Todas las pruebas se realizarán en órbita baja, ya que es donde se desarrolla la misión QBITO.

Esta tesis tiene, por lo tanto, como principal objetivo el estudio del uso de algoritmos basados en lógica difusa para el control de actitud en satélites en órbita baja. Este objetivo puede, a su vez, desglosarse en los siguientes:

- (i) Creación de un entorno de simulación de dinámica orbital. Este simulador será el que nos permita estudiar la respuesta de los algoritmos de control que se diseñen.

- (ii) Diseño de un algoritmo basado en lógica difusa para usarlo en el control de actitud del satélite.
- (iii) Diseño de un control clásico de actitud para el mismo satélite.
- (iv) Comparación del funcionamiento de ambos controladores mediante el empleo de simulaciones numéricas.
- (v) Implementación de un control de actitud embarcado basado en lógica difusa para usarlo de manera experimental a bordo del satélite QBITO como integrante del proyecto QB50. Este módulo debe englobar tanto el algoritmo de control, como el resto de módulos necesarios para el correcto funcionamiento del mismo, con especial mención al módulo de propagación de órbita y al de determinación de actitud.

1.4. Estructura de la tesis.

Esta tesis se estructura en doce capítulos. El primero es la introducción en la que esta sección se engloba.

En el segundo capítulo se hace un repaso de lo que representa una misión espacial así como al entorno en el que se desarrolla.

En el tercer capítulo se explican conceptos importantes para la tesis como el concepto de actitud, su determinación y el subsiguiente control.

En el capítulo cuarto se exponen sucintamente las bases de la teoría de control necesarias más adelante.

El quinto capítulo profundiza en el satélite QBITO y en su misión como parte del proyecto QB50. Se ha hecho especial énfasis en el subsistema de actitud por ser éste el de mayor interés para esta tesis.

En los capítulos sexto y séptimo se desarrolla el proceso de diseño, empezando por las herramientas de soporte necesarias y acabando con los algoritmos de control propiamente dichos.

En los capítulos octavo y noveno se detallan las simulaciones hechas para el estudio de ambos controles. En estos dos capítulos también se explica el procesado de los datos obtenidos y las conclusiones alcanzadas.

El capítulo 10 se detalla el trabajo realizado para desarrollar el código embarcado haciendo un especial énfasis en todos los módulos específicos de las operaciones orbitales.

Como último paso antes de las conclusiones, que se detallan en el capítulo duodécimo, en el capítulo undécimo se explica todo lo relacionado con las operaciones en órbita.

2. El entorno espacial y las misiones espaciales

En este capítulo se definen y clasifican algunos de los conceptos básicos necesarios para el desarrollo de esta tesis. En concreto se va a ver como catalogar los satélites y las distintas misiones espaciales más comunes. Después se va a estudiar el entorno espacial, así como sus implicaciones tanto en el satélite como en su órbita. Por último, se va a hacer una introducción al control de actitud de un satélite, empezando por la definición y representación de la misma, hasta su estimación y posterior control.

2.1. Clasificación de satélites

Un satélite es todo objeto que orbita alrededor de otro cuerpo, llamado primario, por métodos puramente inerciales (i. e. sin propulsión). Existen multitud de maneras de catalogar a un satélite, siendo las más comunes por su tipo de órbita, por su tamaño o por su misión principal.

2.1.1. Clasificación por el tipo de órbita

Si se usa el criterio del tipo de órbita, éstas se pueden catalogar por su tamaño, forma u orientación tridimensional con respecto al cuerpo primario. La primera clasificación se hace en función de la altura de la órbita:

- Órbitas bajas (LEO por sus siglas en inglés: *Low Earth Orbit*): alturas de hasta 1.000 km sobre la superficie.
- Órbitas medias (MEO por sus siglas en inglés: *Medium Earth Orbit*): alturas desde los 1.000 km hasta los 36.000 km.
- Órbitas altas (HEO por sus siglas en inglés: *High Earth Orbit*): alturas mayores que 36.000 km. La misma abreviatura puede referirse a órbitas de gran excentricidad.
- Órbitas Geosíncronas (GEO por sus siglas en inglés: *Geosynchronous Earth Orbit*): altura de aproximadamente 36.000 km.

En la figura 2.1 se puede observar la distribución espacial de las órbitas clasificadas según la altura de las mismas como se acaba de describir.

Figura 2.1. Esquema de clasificación de órbitas según su altura.

Todas ellas pueden, a su vez, ser circulares o elípticas (en caso de ser hiperbólicas o parabólicas, el satélite acabaría escapando del cuerpo primario por lo que dejaría de estar referido a éste).

Una última clasificación puede hacerse según su orientación tridimensional:

- Ecuatoriales, cuando la inclinación (el ángulo que forma la órbita con el plano principal del sistema de referencia, casi siempre el ecuador) es 0° (la órbita está contenida en el plano principal).
- Directas, cuando la inclinación es menor que 90° .
- Polares, cuando la inclinación es cercana a 90° .

- Retrógradas, cuando la inclinación es mayor que 90° .

2.1.2. Clasificación por tamaño

Si se clasifican los satélites por tamaño, la definición estándar los cataloga según la siguiente lista:

- Grandes: masa mayor que 1.000 kg.
- Medianos: masa entre 500 y 1.000 kg.
- Pequeños: masa inferior a 500 kg.
 - Minisatélite: masa entre 100 y 500 kg.
 - Microsatélite: masa entre 10 y 100 kg.
 - Nanosatélite: masa entre 1 y 10 kg.
 - Picosatélite: masa entre 100 g y 1 kg.
 - Femtosatélite: masa menor que 100 g.

Dado que un CubeSat estándar tiene un peso máximo de 1,33 kg, éstos se catalogan como nanosatélites.

2.1.3. Clasificación por el tipo de misión

Por último, la clasificación por el tipo de misión principal del satélite es la más difusa de las tres, ya que un satélite puede tener más de una carga de pago o no adecuarse bien ésta a la categorización. En general puede establecerse la siguiente clasificación:

- De observación terrestre, casi siempre órbitas LEO.
- De observación espacial, con órbitas que dependen de la misión particular.
- De comunicaciones, típicamente en órbitas MEO y GEO.
- De demostración técnica que pueden estar en cualquier tipo de órbita.
- De navegación, típicamente en órbitas MEO y GEO.
- Militares.

2.2. El entorno espacial y las misiones espaciales

A pesar de su cercanía (el inicio del espacio se considera alrededor de los cien kilómetros de la superficie terrestre), el entorno espacial es radicalmente diferente a todo lo encontrado en la Tierra (Tribble, 2003) (Sidi, 2000) (Wertz, 1992). El entorno espacial es extremadamente inhóspito, tanto para la vida como para la tecnología. El principal factor diferencial es el vacío en el que se mueve el satélite y la casi completa ausencia de fricción que conlleva. Este hecho puede tener, *a priori*, puntos positivos ya que sin rozamiento no existen pérdidas energéticas y, por tanto, la gran mayoría de órbitas se pueden considerar eternas a efectos energéticos. Pero también tiene puntos negativos, como, por ejemplo, que hay que idear métodos para moverse en ese medio sin fricción. Además, existe el grave problema de la basura espacial (Belk, 1997): un satélite, cuando deja de estar operativo sigue manteniendo su misma órbita hasta que algún agente externo le haga salir (figura 2.2).

Figura 2.2. Representación, no a escala, del problema de la basura espacial, principalmente en órbitas LEO y GEO donde se observa una preocupante acumulación de elementos. Crédito: Agencia Espacial Europea.

Además del vacío, aunque relacionado con él, en el entorno espacial también supone un grave problema el control térmico (Gilmore, 2002). Al no existir un medio circundante que homogenice térmicamente, las temperaturas en el espacio varían dentro de unos márgenes mucho más extremos. Cuando el satélite se encuentra en la zona iluminada de la órbita

(bajo iluminación solar directa) está recibiendo mucha energía en aproximadamente la mitad de las caras, mientras que el resto están radiando al vacío. A su vez, cada vez que la órbita pasa por la zona de sombra generada por la Tierra, la cantidad de energía que recibe se ve considerablemente menguada. Por lo tanto, mantener una temperatura de operación aceptable para los componentes del satélite se convierte en todo un reto de ingeniería.

Finalmente, si la órbita es suficientemente alta como para cruzar o estar fuera de los cinturones de Van Allen (Van Allen, 1959), el satélite está sometido a fuertes dosis de radiación que pueden afectar a sus componentes, bien de manera puntual, o bien de manera permanente.

En relación con el medio se observa que existen una serie de perturbaciones ambientales que afectan tanto a la órbita como a la actitud del satélite.

(i) Fuerza y par causados por el rozamiento debido a la atmósfera residual.

Este efecto está presente sólo en órbitas LEO bajas (por debajo de 600 km aproximadamente). El rozamiento con la atmósfera residual todavía presente a esas alturas provoca una pérdida de energía cinética (asociada a su vez a una pérdida de energía potencial). Esto conduce a un descenso en la altura de la órbita y, por ende, a un incremento en el rozamiento aerodinámico. Se puede observar cómo este proceso tiene una forma exponencial y acaba, si no se actúa evitándolo, con el satélite reentrando en la atmósfera y cayendo a tierra. Como se detalla en la sección 5.3.1, la misión QBITO sigue una órbita LEO baja por lo que ésta es la perturbación principal, afectando tanto a la órbita como a la actitud del satélite.

(ii) Efectos provocados por la asimetría en el campo gravitatorio terrestre.

Aunque en casi todos los casos se considera al cuerpo primario como una esfera perfecta o una masa puntual, los cuerpos reales no son ni masas puntuales ni esferas perfectas. En el caso de la tierra, la principal asimetría está causada por el achatamiento terrestre. Éste es debido a la rotación sobre su eje que provoca que el radio ecuatorial sea mayor que el terrestre. Este fenómeno recoge su efecto en el parámetro llamado J_2 (Stacey, 2008). Este nombre se debe a que el efecto del achatamiento terrestre en el campo gravitatorio terrestre está recogido, principalmente, en el segundo término de un desarrollo en serie de armónicos esféricos que representan el potencial gravitatorio terrestre. Este fenómeno (el achatamiento terrestre) causa que, principalmente en órbitas bajas, la órbita pueda llegar a precesar incluso más de 10° por día dependiendo de las características de la órbita.

(iii) Efectos provocados por la influencia gravitatoria del resto de cuerpos del universo, principalmente del resto del sistema solar.

Una simplificación que suele hacerse habitualmente es la consideración de que solo existen dos cuerpos en el problema en cuestión: la Tierra o el cuerpo primario y el satélite. Sin embargo, esto nunca es así dado que el resto de cuerpos celestes siempre van a existir y, por lo tanto, interferir gravitatoriamente. De todas maneras la aproximación de considerar únicamente dos cuerpos suele ser buena en la gran mayoría de escenarios dado que, comparativamente, la influencia del resto de cuerpos es pequeña con respecto a las fuerzas principales implicadas. Sin embargo, en órbitas altas, con especial mención en las órbitas GEO, puede llegar a ser la perturbación dominante, superando los efectos causados por el achatamiento terrestre (efecto del J_2).

(iv) Fuerza y par causados por la presión de radiación solar.

Esta fuerza está causada por la presión que causa la radiación electromagnética al incidir sobre una superficie. Como la luz solar incide únicamente sobre la mitad del satélite, esto provoca una asimetría de presiones. Este efecto solo suele ser importante en satélites muy ligeros y grandes (las llamadas velas solares) por lo que no merece la pena entrar en detalle por ser un efecto marginal en la gran mayoría de satélites.

(v) Par causado por la interacción del campo magnético terrestre con el posible campo magnético residual del satélite.

Todo satélite tiene un cierto dipolo magnético residual, aunque este suele ser muy pequeño. Sin embargo, en ausencia de otras fuerzas, y en presencia del campo magnético terrestre, el dipolo residual tiende a alinear al satélite con el campo magnético local, alterando la actitud del satélite. Este proceso se puede asimilar a una brújula, donde el satélite con su dipolo residual se comportaría como la aguja imantada.

(vi) Par causado por la no uniformidad másica del satélite (efecto de marea).

En un satélite, cada punto sufre una aceleración gravitatoria distinta al resto. Esta fuerza viene determinada por la ubicación instantánea del punto infinitesimal dentro del campo gravitatorio así como por sus propiedades másicas. Este efecto, que solo tiene impacto en la actitud del satélite y no en la órbita, provoca que se alineen el eje de mayor momento de inercia del satélite con la vertical local. La vertical local es la dirección marcada por la línea imaginaria que une el centro de la Tierra con el centro de masas del

satélite. El proceso se podría asemejar a la dinámica de un “tetratión” espacial.

En la lista anterior aparecen todas las perturbaciones posibles sin atender a su efecto: alteración de la altura, forma u orientación de la órbita o pares de perturbación que afectan a la actitud del satélite. Los efectos causados en la actitud del satélite van a ser estudiados más en detalle a continuación en la sección 2.2.1. Por otro lado, dado que en este estudio se ha utilizado la misión de QBITO como objetivo, y éste no tiene actuadores capaces de afectar a la órbita del satélite, no se ha profundizado ni en el efecto sobre la órbita ni en el control de la misma.

2.2.1. Efecto de las perturbaciones en la actitud del satélite

La figura 2.3 muestra los pares de perturbaciones esperables para un CubeSat similar a QBITO en función de la altura de la órbita. Como se observa en la figura 2.3, en órbitas LEO bajas el par más importante suele ser el aerodinámico. Al aumentar la altura, el par causado por dipolo magnético gana relevancia ya que la densidad atmosférica disminuye rápidamente con la altura. Éste par es el más importante en las órbitas LEO altas así como en las MEO y algunas HEO. Aproximadamente a partir de las órbitas GEO (que aparecen en el extremo derecho de la figura 2.3), el par más relevante es el causado por la presión solar. Por último, el par por gradiente gravitatorio es significativo solo en satélites con geometrías especiales (v. g. que cuenten con booms desplegados), siendo despreciable para el resto de casos.

A continuación se analiza más en detalle cada uno de los pares de perturbación. Se empieza por el par aerodinámico ya que es el más importante para la misión de QBITO. Éste se debe a la no alineación del centro de presiones con el centro de masas del satélite, causando que la fuerza de la resistencia aerodinámica provoque a su vez un par de perturbación. Este par puede estabilizar o desestabilizar el satélite dependiendo de la geometría y la actitud del satélite.

La segunda perturbación a considerar es el par causado por el dipolo magnético residual del satélite al interactuar con el campo magnético terrestre. El dipolo residual está causado por la magnetización de los componentes del satélite y es desconocido *a priori*. Por lo tanto se suelen tomar valores característicos durante la etapa de diseño hasta tener el

satélite final construido y analizado. Este par de perturbación intenta alinear el dipolo magnético residual del satélite con el campo magnético terrestre.

Figura 2.3. Pares de perturbación esperables para un CubeSat según la altura de la órbita para las órbitas típicas.

Siguiendo en órbitas bajas, la siguiente perturbación a considerar es la causada por el par por gradiente gravitatorio. Esta perturbación está causada por el hecho de que el campo gravitatorio decrece con la distancia, provocando que la parte más pesada del satélite tienda a apuntar hacia el centro de la Tierra. Este efecto causa que tiendan a alinearse el eje de mayor momento de inercia con la vertical local. En algunos casos se desea maximizar esta característica por lo que se hace uso de elementos desplegables (v. g. una masa que se aleja del cuerpo del satélite por medios mecánicos) que incrementan uno de los momentos de inercia sensiblemente respecto a los otros dos.

Por último, el par producido por la presión solar tiene poca influencia en órbitas bajas, su influencia es importante en órbitas HEO como la GEO y superiores. En este caso, como en el del par aerodinámico, el par está causado por la no alineación del centro de presiones y el centro de masas. Aunque en valor absoluto este par sea pequeño, a lo largo del tiempo puede tener efectos significativos cuando el resto de pares no están presentes o son despreciables.

3. Determinación y control de órbita y actitud en satélites

La actitud de un satélite es su orientación tridimensional con respecto a un sistema de referencia arbitrario. El control de actitud tiene como objetivo el orientar tridimensionalmente al satélite de forma que cumpla con las especificaciones requeridas de la misión. Debido a la inmediatez del control, por normal general, todas las etapas del control de actitud suelen ser desarrolladas a bordo aunque en algunos escenarios concretos las partes más pesadas del cómputo pueden ser realizadas en tierra y transmitidas al satélite vía radio.

En un escenario ideal, en el que no hay perturbaciones, una vez alcanzada la actitud deseada, el satélite permanecería permanentemente en ella pudiendo ser obviado el control de actitud. Sin embargo, como ya se ha mencionado en la sección 2.2.1, existen diferentes perturbaciones que pueden afectar a la actitud del satélite por lo que es imprescindible contar con algún tipo de control de actitud.

3.1. Representación de la actitud

Como se acaba de mencionar, la actitud de un satélite respecto a un sistema de referencia dado es la orientación tridimensional de unos ejes fijos al satélite respecto a dicho sistema de referencia. En el Anexo A se explican en detalle los principales sistemas de referencia empleados a lo largo de la tesis. Existen varias maneras de representar la actitud de un satélite, cada una con sus pros y sus contras. A continuación se van a exponer las principales representaciones de actitud usadas en misiones espaciales.

3.1.1. Ángulos de Euler.

Esta representación es la más intuitiva, pues codifica la actitud mediante tres giros consecutivos alrededor de unos ejes fijos al satélite (llamados ejes cuerpo o BRF por sus siglas en inglés: *Body Reference Frame*, y que se denotan por $\mathbf{X}_B\mathbf{Y}_B\mathbf{Z}_B$, ver la sección A.4 del Anexo A). La actitud suele referenciarse a unos ejes o bien inerciales o bien que se mueven con el satélite. Debido a las características de la misión de QBITO (sección 5.3.1) la actitud se ha expresado siempre respecto a unos ejes fijos a la órbita (sistema

de referencia orbital u ORF por sus siglas en inglés: *Orbit Reference Frame* y que se denotan por $\mathbf{X}_O\mathbf{Y}_O\mathbf{Z}_O$, ver la sección A.3 del Anexo A).

No existe una única manera de efectuar esta descomposición (convertir una actitud arbitraria en tres giros consecutivos) sino que dependerá del problema que se estudie. Para denominar las descomposiciones se utiliza una secuencia de tres dígitos que representa el orden y los ejes sobre los que se efectúan los giros. Por ejemplo, en dinámica de sólidos rígidos se suele utilizar una descomposición del tipo 3-1-3 (giro alrededor del eje $\mathbf{Z}_B$, giro alrededor del eje $\mathbf{X}_B$ y otro giro en $\mathbf{Z}_B$) dado que ésta permite simplificar las ecuaciones estudiadas. Por el contrario, en vehículos aeroespaciales tiende a usarse una descomposición del tipo 3-2-1 y es la que se usará en esta tesis. Esta descomposición representa la actitud como un giro respecto al eje $\mathbf{Z}_B$, un giro respecto al $\mathbf{Y}_B$ y, por último, un giro respecto al eje $\mathbf{X}_B$. Como ejemplo, la figura 3.1 representa una actitud de $-\left[\frac{\pi}{9}, \frac{\pi}{3}, \frac{\pi}{5}\right]$ radianes entre los ejes ORF y los ejes BRF.

Figura 3.1. Representación de la actitud del sistema $\mathbf{X}_B\mathbf{Y}_B\mathbf{Z}_B$ respecto al $\mathbf{X}_O\mathbf{Y}_O\mathbf{Z}_O$. *Roll*, *pitch* y *yaw* son los nombres en inglés para los tres ángulos de Euler: balanceo, cabeceo y guiñada; $\mathbf{v}$ y δ son el vector y el ángulo respectivamente que representarían el giro mediante cuaterniones (sección 3.1.2).

Este tipo de representaciones son útiles para visualizar la situación tridimensional pero no para su uso en computación ya que exigen múltiples operaciones trigonométricas (muy costosas computacionalmente). Además, en toda representación de actitud con solo 3 parámetros (como los ángulos de Euler) existen singularidades cuya gestión complica significativamente el código que se encargue de la actitud.

3.1.2. Cuaterniones o parámetros de Euler-Rodrigues.

Como se acaba de mencionar, toda representación con solo tres parámetros presenta singularidades. Para evitarlo se escogen parametrizaciones que, al incorporar redundancias evitan dicha singularidad. Las representaciones más frecuentes son los cuaterniones, que se explican en esta sección y la matriz de giro (ver sección 3.1.3). La representación mediante cuaterniones está basada en 4 parámetros y facilita el cómputo de la determinación de actitud siendo además bastante eficiente computacionalmente. Por el contrario la interpretación física y su visualización tridimensional es bastante compleja. Los cuaterniones pueden ser representados como un giro arbitrario, δ , alrededor de un eje tridimensional también arbitrario, $\mathbf{v}$, como se puede observar en la figura 3.1 junto a la representación del giro mediante ángulos de Euler. Para pasar del conjunto ángulo-vector al cuaternión se usa la ec. (3.1).

$$\mathbf{q} = \begin{bmatrix} v_x \sin\left(\frac{\delta}{2}\right) & v_y \sin\left(\frac{\delta}{2}\right) & v_z \sin\left(\frac{\delta}{2}\right) & \cos\left(\frac{\delta}{2}\right) \end{bmatrix} = [\vec{\mathbf{q}} \quad q_0] \quad (3.1)$$

En esta ecuación se ha representado el cuaternión, tanto descompuesto en su representación ángulo-vector como en su representación vector-escalar ($\vec{\mathbf{q}}$ y q_0 respectivamente). La primera es útil para visualizar el giro representado, mientras que la segunda se suele utilizar en desarrollos matemáticos. Por ejemplo, la rotación representada en la figura 3.1 se obtiene para los valores siguientes:

$$\mathbf{q}_{\text{BO}} = [0.2951 \quad 0.4218 \quad 0.3461 \quad 0.7843] \rightarrow \\ \mathbf{v} = [0.4758 \quad 0.6799 \quad 0.5579]; \delta \approx -1.34 \text{ rad}$$

Donde $\mathbf{q}_{\text{BO}}$ representa el cuaternión entre los ejes BRF y ORF.

3.1.3. Matriz de giro o de cosenos directores.

Esta matriz representa el triedro de un sistema de referencia en coordenadas de otro sistema de referencia diferente. Tiene el inconveniente de que 6 de los 9 parámetros de la matriz son redundantes por lo que se incrementa el cómputo no útil. En contrapartida, esta representación no tiene singularidades y la composición de giros es inmediata pues se calcula como un producto matricial. En forma general puede escribirse como se muestra en la ec. (3.2) usando la notación de la ec. (3.1). Además se ha añadido el valor numérico de dicha matriz para la rotación representada en la figura 3.1.

$$\mathbf{R} = \begin{bmatrix} \cos \delta + v_x^2(1 - \cos \delta) & v_x v_y(1 - \cos \delta) - v_z \sin \delta & v_x v_z(1 - \cos \delta) - v_y \sin \delta \\ v_x v_y(1 - \cos \delta) - v_z \sin \delta & \cos \delta + v_y^2(1 - \cos \delta) & v_z v_y(1 - \cos \delta) - v_x \sin \delta \\ v_x v_z(1 - \cos \delta) - v_y \sin \delta & v_z v_y(1 - \cos \delta) - v_x \sin \delta & \cos \delta + v_z^2(1 - \cos \delta) \end{bmatrix} \quad (3.2)$$

$$\rightarrow \mathbf{R} = \begin{bmatrix} 0.4045 & 0.7920 & -0.4573 \\ -0.2939 & 0.5861 & 0.7550 \\ 0.8660 & -0.1710 & 0.4698 \end{bmatrix}$$

En resumen, la actitud representada en la figura 3.1 se escribiría en cada una de las representaciones anteriormente descritas como:

- $[\varphi, \theta, \psi] = -\left[\frac{\pi}{9}, \frac{\pi}{3}, \frac{\pi}{5}\right]$
- $\mathbf{q}_{B0} = [0.2951 \quad 0.4218 \quad 0.3461 \quad 0.7843] \rightarrow \mathbf{v} = [0.4758 \quad 0.6799 \quad 0.5579]; \delta \approx -1.34 \text{ rad}$
- $\mathbf{R} = \begin{bmatrix} 0.4045 & 0.7920 & -0.4573 \\ -0.2939 & 0.5861 & 0.7550 \\ 0.8660 & -0.1710 & 0.4698 \end{bmatrix}$

Se puede observar que la representación usando ángulos de Euler permite visualizar de forma gráfica fácilmente el estado del problema mientras que con el resto de ellas intentar visualizar el problema se hace prácticamente imposible.

3.2. Control de actitud

El algoritmo de control es el cerebro de todo el subsistema de actitud. Éste toma la actitud actual del satélite (estimada por el módulo de determinación como se detalla más adelante en la sección 3.3) y calcula los comandos necesarios que deberán efectuar los actuadores para conseguir que

el satélite alcance la actitud escogida. En el capítulo 4 se va a hacer un estudio pormenorizado de la teoría de los algoritmos de control utilizados en esta tesis. Además, en el capítulo 7 se detalla el proceso seguido para el diseño y calibración de los algoritmos específicos usados en esta tesis.

El proceso de control puede ser visualizado esquemáticamente en la figura 3.2.

Figura 3.2. Esquema general de control. “Comp” es la abreviatura de comparador.

La referencia es el objetivo que se quiere conseguir, éste puede ser escalar o vectorial, estacionario o dinámico. En el caso aquí considerado, la referencia es la actitud deseada del satélite. Dentro del bloque de "Sensores" se engloban tanto los sensores del satélite como cualquier proceso de filtrado y posprocesado que se considere necesario para el control. Además en este bloque también se engloba el algoritmo de determinación de actitud. Esta actitud se compara con la de referencia para estimar el error.

El error calculado se introduce en el bloque de control, cuya misión es proporcionar la actuación necesaria para alcanzar la actitud establecida en la referencia. El bloque de "Control" puede tener cualquier algoritmo de control que se estime conveniente. Los controles típicos en sistemas espaciales suelen estar basados en el control clásico PID (que se detallará a continuación en la sección 4.1) o en controles más avanzados como el LQR o en la asignación de autoestructuras (Crespo, 2011). Por último, existe toda una rama de controladores que está empezando a hacer su aparición en algoritmos embarcados en órbita como son los controles inteligentes, el uso de redes neuronales, algoritmos genéticos o algoritmos de lógica difusa. Éste último se explica más en detalle en la sección 4.3 al ser el que se va a implementar en esta tesis junto con un control clásico de tipo PID.

Por último, la salida del controlador se envía a los actuadores que afectarán a la planta (i. e. el satélite) modificando su actitud. La planta representa el problema físico que debe ser controlado. En el caso de este

estudio, la planta es la dinámica de rotación del satélite en el cuasi-vacío del espacio, sometido a perturbaciones ambientales y a pares de control.

En las secciones siguientes de este capítulo se describe en detalle cada uno de los componentes del algoritmo completo de control de actitud del satélite.

3.3. Determinación de actitud

Antes de poder aspirar a controlar la actitud de un satélite, éste tiene que ser capaz de determinarla de manera autónoma, precisa y rápida. Como paso previo al proceso de determinación de actitud, y por ende del control de actitud, es necesario saber la posición instantánea del satélite respecto a unos ejes inerciales (ECI por sus siglas en inglés, *Earth Centered Inertial*, detallados en el Anexo A, sección A.1). Esto se suele hacer, bien por medición directa (i. e. lectura de GPS) cuando es posible o bien por estimación (i. e. mediante propagadores orbitales) cuando el uso del GPS no es posible. Conocer la posición instantánea es necesario porque para determinar la actitud se necesitan una serie de vectores de referencia medidos en el sistema de referencia ORF, que dependen de la posición orbital del satélite en cada instante. Estos vectores representan la lectura que proporcionarían los sensores del satélite si éste tuviese una actitud predefinida (típicamente cuando coinciden los ejes BRf con los ejes ORF). Estas estimaciones se hacen mediante modelos físicos, como se explica en la sección 10.1.3.

Una vez conocida la posición orbital del satélite, se procede a la determinación de su actitud. El proceso de determinación de la actitud de un satélite consiste en utilizar los sensores disponibles a bordo, estimar la lectura que deberían dar para una actitud dada (v. g. en QBITO se usa la actitud nominal: es decir cuando los ejes BRf coinciden con los ejes ORF) y comparar ambos pares de valores (las lecturas estimada y las lecturas reales) para estimar el error en la actitud. Este proceso se ve representado en la figura 3.2 por el comparador (abreviado como "Comp"). Dado que sobre este proceso se construye todo el control de actitud es importante que sea robusto, fiable y rápido para que dé tiempo a ejecutar el algoritmo de control.

Existen múltiples técnicas y algoritmos para realizar la comparación entre las lecturas de los sensores y las estimaciones hechas a bordo para obtener la actitud instantánea del satélite. Los más famosos son los algoritmos TRIAD (Black, 1964), QUEST (Shuster, 1978) (Shuster, 1981) y

Qmethod (Keat, 1977). El control de actitud de QBITO tiene implementados tanto el TRIAD como el Qmethod aunque se basa principalmente en éste último ya que al admitir más de dos medidas (QBITO puede proporcionar hasta cuatro lecturas de sensores diferentes) puede proporcionar estimaciones más precisas. A su vez, esta actitud estimada puede ser filtrada, teniendo en cuenta las estimaciones previas, para obtener una predicción en actitud mejor. Entre los filtros más habituales están el RE-QUEST, el filtro alfa-beta o el filtro de Kalman (Kalman, 1960), siendo éste último el más utilizado y el que se ha implementado en el SW embarcado en QBITO.

Como ya se ha mencionado, una buena precisión de este algoritmo es determinante para el funcionamiento del control de actitud. Los principales factores que pueden afectar a dicha precisión son, principalmente, los sensores usados a bordo del satélite y también aunque en menor medida los algoritmos y modelos.

3.4. Sensores

Como se ha comentado anteriormente, una buena elección de los sensores embarcados es crucial para conseguir un control de actitud preciso. Los sensores habitualmente usados en sistemas en órbita se pueden clasificar principalmente como en la siguiente lista:

- Magnetómetros o sensores de campo magnético.
- Sensores terrestres.
- Sensores solares.
- Sensores estelares.
- Sistemas inerciales (giróscopos).

Las precisiones típicas oscilan desde algunos grados en el caso de los magnetómetros típicos hasta pocos segundos de arco en los sensores estelares más precisos. La precisión de los sistemas inerciales va empeorando con el tiempo por lo que hay que tenerlo en cuenta al hacer la comparación con el resto de sensores. Sin embargo, además de la precisión característica, cada tipo de sensores suele tener otro tipo de restricciones que impiden o complican su uso en determinadas circunstancias. En el caso del magnetómetro, éste deja de ser útil a partir de las órbitas HEO debido a que el campo magnético es muy débil a esas alturas. El resto de sensores, sin contar los inerciales, suelen tener un campo de visión (FOV por sus siglas en inglés: *Field Of View*) que oscilará entre algunos grados y algunas decenas de

grados, limitando la maniobrabilidad de satélite o forzando el uso de varios sensores apuntando en direcciones diferentes para tener lecturas permanentemente. Además, algunos sensores pueden no funcionar en eclipse por lo que habrá que prever alternativas en caso de que la órbita tenga eclipses. Típicamente esto se suele contrarrestar mediante el uso de sensores diferentes o mediante filtros recursivos como el filtro de Kalman extendido que aportan una ventaja inherente al seguir funcionando con precisión razonable a pesar de la ausencia de nuevas medidas.

En el caso de la misión QBITO, no se han embarcado sensores estelares debido a su alto coste y a que la precisión necesaria puede ser alcanzada con el resto de sensores. Además, QBITO también incorpora dos tipos de sensores solares, un juego de sensores omnidireccionales pero de baja precisión y un sensor de sol preciso pero de pequeño FOV. De esto se hablará más en detalle en la sección 5.3.6.

3.5. Actuadores

Las salidas del algoritmo de control son los comandos a los actuadores necesarios para obtener o mantener la actitud de referencia. A continuación se han detallado los diferentes tipos de actuadores típicamente usados en control de actitud de satélites.

El primer actuador que se considera son las ruedas de inercia. Éstas, al rotar, giran el satélite en sentido contrario al de la rotación debido a la conservación de momento angular. Si su punto de operación nominal es alrededor del reposo se llaman ruedas de reacción. Por el contrario si siempre están girando a altas revoluciones y actúan aumentando o disminuyendo esa velocidad se denominan ruedas de momento. Estas últimas tienen la ventaja de que además estabilizan giroscópicamente el satélite en las dos direcciones ortogonales al eje de giro. En ambos casos, las ruedas acumulan momento angular si las perturbaciones tienen componente secular. En esos casos la velocidad de giro de la rueda tiende a alejarse del valor nominal. Cuando esto ocurre, es necesario descargarlas del momento acumulado. Para ello se requiere de un segundo actuador (que no acumule momento sino que lo intercambie con el exterior) que compense la descarga de momento sin afectar a la dinámica del satélite. Este segundo actuador puede ser magnético o propulsivo.

Los controladores magnéticos son, en esencia, imanes que ejercen un par que tiende a alinear el vector dipolo magnético instantáneo con el vector campo magnético terrestre local. Si en vez de imanes se hace uso de electroimanes, el dipolo magnético es modificable. Este método tiene el inconveniente de que el par que se genera depende de la orientación del campo magnético (es siempre ortogonal tanto al dipolo magnético como al campo magnético terrestre), lo que puede no ser conveniente para la actuación deseada.

Por último, los controladores mediante propulsión son los más versátiles pero tienen como inconveniente que necesitan de combustible para su funcionamiento y su mayor complejidad. Este método consiste en expulsar gas a alta velocidad de manera asimétrica para producir momento pero sin producir un empuje neto. Como ya se ha mencionado, el principal inconveniente de este método es que requiere de combustible (típicamente hidracina), el cual, una vez acabado finaliza la misión.

Habitualmente los satélites suelen llevar varios tipos de actuadores. La elección final depende de la misión a desarrollar, así como de los requisitos de actitud de la misma.

3.6. Métodos de estabilización pasivos

Además del uso de un control activo como se ha explicado antes, existen métodos que estabilizan al satélite en una serie de actitudes determinadas. Los métodos pasivos tienen la ventaja de no requerir ni potencia ni combustible (ni controlador en muchos casos). En contrapartida suelen tener precisiones mucho peores y la actitud del satélite suele estar determinada por parámetros físicos. Los dos sistemas clásicos de estabilización pasiva son por alineación magnética y por gradiente de gravedad.

El método de alineación magnética (Danchik, 1998) tiene un funcionamiento parecido a una brújula: estando el satélite magnetizado, éste intenta alinearse con el campo magnético local. Este método ha sido usado en ciertas misiones de bajo presupuesto ya que es la manera más simple y económica de estabilizar un satélite.

El método de estabilización por gradiente de gravedad (Robertson, 1958) alinea el eje de mayor momento de inercia según la vertical local. Para conseguir mejores resultados se suele favorecer este fenómeno incrementando

artificialmente uno de los momentos de inercia mediante un boom o mediante geometrías muy degeneradas.

En un punto intermedio entre los métodos pasivos y el control activo estaría la estabilización por spin. En este método el satélite giraría sobre sí mismo alrededor de su eje de inercia mayor a una velocidad significativa produciendo una estabilización giroscópica. Es importante destacar que si se aplica esta técnica al eje que no sea el de mayor momento de inercia se puede llegar a perder la misión, como ya pasó con el Explorer 1 (Harland, 2005). Este método no entra del todo en los métodos pasivos porque necesita de algún agente externo que acelere el satélite inicialmente.

4. ALGORITMOS DE CONTROL

4.1. Introducción

Un control realimentado genérico se puede esquematizar como se ha visto en la figura 3.1. En ese esquema un algoritmo de control ajusta una o más variables de la planta mediante actuadores y recibe realimentación del estado de la planta mediante sensores. Una planta puede ser cualquier proceso del que se desee establecer su estado con una cierta precisión, desde la temperatura del agua en una bañera hasta la actitud de un satélite de observación astronómica.

En este capítulo se van a estudiar en detalle dos aproximaciones diferentes al problema del control. La primera es el uso de algoritmos clásicos, en concreto un algoritmo de tipo PID. En la segunda parte se va a explicar en detalle el funcionamiento de un control basado en lógica difusa. En el capítulo 7 se detalla el diseño de ambos controladores para el problema particular del control de actitud de un CubeSat en órbita LEO.

4.2. PID

El control PID es un control clásico y probablemente el más extendido en la industria por su fácil comprensión e implementación (Ziegler, 1993) (Åström, 1995).

Un control PID tiene tres actuaciones combinadas, todas ellas basadas en una medida del error entre la referencia propuesta y el estado de la planta. Estas tres actuaciones o ramas son la proporcional, que se basa en el valor del error directamente, la rama integral, que se basa en el error acumulado a lo largo del tiempo, y la rama derivativa que se basa en la tasa de variación temporal del error. El funcionamiento de un control de tipo PID se puede expresar lingüísticamente y de manera poco formal con el siguiente conjunto de afirmaciones: "cuanto más lejos estoy del objetivo más fuerte intento llegar a él", "cuanto más tiempo pase lejos del objetivo más fuerte voy a intentar llegar a él" y por último "si me estoy moviendo voy a intentar frenarme". Es obvio que las tres actuaciones descritas no siempre van a ser coherentes entre sí; de hecho, es un comportamiento deseado ya que consigue llegar a la solución deseada y amortiguar la respuesta del proceso.

Las leyes lingüísticas descritas previamente se pueden esquematizar como se observa en la figura 4.1. Se observa como la señal de control es procesada antes de su uso obteniendo la tasa de variación del error así como el error acumulado a lo largo de la historia del control. Además, las ganancias K_p , K_i y K_d (ganancia proporcional, integral y derivativa respectivamente) son las que determinan el comportamiento del sistema y deben ser ajustadas para la planta en cuestión. En caso de que la señal de error fuese multidimensional y que las señales fuesen independientes el esquema se repetiría para cada una de las dimensiones. Si las señales no son independientes el proceso se complica dado que habría que tener en cuenta correlaciones y efectos cruzados entre dimensiones que no se han detallado aquí por simplicidad.

Figura 4.1. Esquema de un control de tipo PID.

Como ya se ha mencionado, cambiando las ganancias K_p , K_i y K_d se modifica la respuesta del control. A modo de ejemplo se representan las reacciones de un control sencillo ante una señal de referencia de tipo escalón cuando se modifican las tres ganancias por separado. En todos los ejemplos se usa la misma planta. Además, siempre se parte de un control ya ajustado para ver el efecto de incrementar o reducir la ganancia en cuestión (K_p , K_i o K_d). En el control de referencia (curvas verdes) los valores de las ganancias son todos unitarios.

Figura 4.2. Cambio en la respuesta de un control PID al verse modificadas sus ganancias. La gráfica superior izquierda muestra el efecto de modificar K_p . La gráfica superior derecha muestra el efecto de modificar K_d . Por último, la gráfica inferior muestra el efecto de modificar la ganancia K_i . Fuente: Wikipedia.

Se observa en la figura 4.2 como al incrementar la ganancia proporcional K_p se consiguen respuestas más ágiles pero que precisan de más tiempo para estabilizarse. El efecto de incrementar la ganancia integral K_i es similar al de incrementar la ganancia proporcional. Sin embargo, al incrementar K_i se observa como la respuesta es mucho más abrupta y tiene un exceso de *overshoot* importante (línea negra en la gráfica inferior de la figura 4.2). Por último, incrementar la ganancia derivativa K_d amortigua la respuesta del control. Sin embargo, si se aumenta mucho dicha ganancia, puede llegar a hacerse inestable el control dado que se amplificaría demasiado el ruido en la señal de entrada y éste alteraría el control.

4.3. Lógica difusa

La lógica difusa es una extensión de la lógica tradicional (o booleana) que permite reproducir los patrones del pensamiento humano y representar matemáticamente la incertidumbre y la vaguedad proporcionando herramientas formales para su tratamiento (Ross, 2010). A la hora de hablar del tratamiento de la incertidumbre, conviene aclarar que los conceptos de lógica difusa (o *fuzzy* en inglés) y probabilidad son completamente diferentes. La probabilidad trata la incertidumbre estocástica, es decir, la incertidumbre de que ocurra un determinado evento. Por ejemplo, en un juego de lanzamiento de dardos, la afirmación: “La probabilidad de acertar en el blanco es 0.8”, la incertidumbre surge de si el blanco será alcanzado o no.

Por otra parte, la lógica difusa trata la incertidumbre basada en el lenguaje de comunicación humano. Por ejemplo supóngase la siguiente afirmación,: “Los pacientes con malaria muestran en el 70% de los casos fiebre alta, el 20% tiene fiebre media y el 10% restante no presenta síntomas de fiebre”, y que, en función de si la fiebre es alta, media o no tiene fiebre, el paciente debe recibir un tratamiento u otro. Si se quiere implementar esta afirmación en un sistema, el problema surge cuando se trata de definir qué es una fiebre alta, ya que ésta es una afirmación subjetiva o de incertidumbre léxica.

El concepto de lógica difusa fue creado por Lofti A. Zadeh (Zadeh, 1965) (Zadeh, 1973) (Zadeh, 1975). En su propuesta Zadeh presentó la lógica difusa como una forma de procesamiento de información en la que los elementos o datos podrían tener un grado de pertenencia parcial a conjuntos, a diferencia de la lógica Booleana, en la que se entiende que el grado de pertenencia de un elemento a un conjunto solo puede ser 1 (si pertenece al conjunto) o 0 (si no pertenece).

A continuación se va a ilustrar esta idea con un ejemplo: Imagínese que se quiere abrir un grifo y que el agua tenga una temperatura "templada" para una ducha. Al abrir el grifo, ésta sale "fría" por lo que se pone el mando en "caliente". Al notar que la temperatura está subiendo se pone el grifo en "templada". Sin embargo por razones de inercia, la temperatura del agua acaba saliendo primero “fría”, después "caliente" y finalmente "templada".

Para cualquier ser humano el procedimiento descrito es extremadamente sencillo. Sin embargo es complicado cuantificar los valores "templada", "fría" o "caliente" de manera precisa. La lógica booleana

establecería tres conjuntos: “Fría”, “Templada” y “Caliente”, definidos mediante rangos de valores. Un ejemplo podría ser usar tres valores como los de la figura 4.3.

Figura 4.3. Clasificación de la temperatura del agua mediante lógica booleana.

Por ejemplo en la figura 4.3 se considera que el agua está "Fría" si la temperatura va desde cero grados hasta 24.99°, "Templada" si va desde 25° hasta 29.99° y "Caliente" desde 35° hasta 50°. Sin embargo, esta cuantificación establece que 24.9° es "Fría" pero 25° es "Templada" cuando ambos valores son prácticamente indistinguibles para un ser humano normal. De hecho, una persona normal diría que 24° tiene mucho de "templada" y un poco de "fría". En otras palabras, haría la frontera entre ambos términos difusa. Esa es la base de la lógica difusa: catalogar las entradas en conjuntos difusos que se superponen unos a otros. Esto permite sobreponerse a las incertidumbres inherentes de cualquier sistema físico así como tener un comportamiento un poco más "inteligente". Por tanto, el primer paso en la lógica difusa consiste en catalogar la entrada (el dato de la temperatura del agua) en una serie de etiquetas lingüísticas como "Fría", "Templada" o "Caliente" mediante lo que se conoce como funciones de pertenencia. El proceso seguido puede observarse en la figura 4.4. En dicha figura se representa la transformación de los valores de temperatura en etiquetas lingüísticas, de manera que, por ejemplo, para la temperatura de 22.5°, se interpretaría que el grado de pertenencia de dicha temperatura al conjunto "Fría" es del 0.75 y que el grado de pertenencia al conjunto "Templada" es de 0.25. Mientras que en la lógica booleana, siguiendo la definición de conjuntos de la figura 4.3 diría que la temperatura de 22.5° pertenece al conjunto "Fría".

Figura 4.4. Ejemplo de funciones de pertenencia. Se ha llamado a la etiqueta "Templada" como "OK" por razones estéticas. Como referencia se ha calculado la pertenencia de una temperatura de 22.5 grados (0.75 "Fría" y 0.25 "Templada").

Se pueden crear tantas funciones de pertenencia como se requiera, y éstas pueden ser de formas totalmente arbitrarias y solaparse total o parcialmente unas con otras. La única norma es que todas ellas tienen que variar entre 0 y 1. Por lo general suelen tener formas triangulares, trapezoidales o gaussianas. Además, como normal general la superposición solo se da entre funciones de pertenencia inmediatamente adyacentes.

Estas funciones de pertenencia pueden ser diferentes para cada señal de entrada, y puede haber tantas entradas como se necesite cuando se emplea la lógica difusa en control, como se ve más adelante.

4.3.1. Control basado en lógica difusa

Imagínese que se quiere diseñar un sistema de control de la temperatura de una habitación basado en lógica difusa. El proceso de control puede ser representado como en la figura 4.5.

Figura 4.5. Esquema de funcionamiento de un control basado en lógica difusa.

A continuación se describen cada uno de los bloques de este sistema de control.

4.3.1.1. Proceso de *Fuzzificación*

En el primer bloque se ha usado el anglicismo *fuzzificar*, que es un anglicismo adaptado dado que no existe equivalente en español. En este primer bloque se transforman las entradas numéricas (los datos de temperatura de la habitación) en variables lingüísticas, empleando funciones de pertenencia como se hizo en la figura 4.4.

4.3.1.2. Reglas

Una vez que se ha catalogado la señal de entrada en variables lingüísticas se pasa a la etapa de las llamadas reglas de evaluación o de inferencia. El controlador difuso emplea reglas lingüísticas sobre las entradas difusas generadas en el primer bloque.

Las reglas difusas son sentencias del tipo: Si ... Entonces ... (If-Then), que describen la acción a ser tomada en función de la entrada. Por ejemplo:

Si la temperatura es "Fría" (y / o ...) entonces la actuación es "Calentar".

Si la temperatura es "Templada" (y / o ...) entonces la actuación es "Nada".

En estas reglas se observa cómo la actuación de la salida se expresa también mediante las variables lingüísticas: “Enfriar”, “Nada” y “Calentar”.

Dentro del paréntesis (y / o ...) se pueden incluir tantas condiciones como se requiera, pudiendo ser de diferentes señales de entrada. Por ejemplo, si se desea controlar la temperatura y la humedad de la habitación, habría que *fuzzificar* las dos entradas, definiendo también funciones de pertenencia para la humedad y reglas de la forma:

Si la temperatura es "Fría" y la humedad es “Alta” entonces la actuación es "Calentar".

En función de los grados de pertenencia de las variables de entrada a cada etiqueta lingüística se obtienen los grados de pertenencia de las variables de salida a las etiquetas correspondientes. Cuando solo hay una variable de entrada y una de salida en la regla lo normal es “traspasar” el grado de pertenencia de la entrada a la salida, de manera que la salida del ejemplo tendría un grado de pertenencia de 0.75 aplicado al conjunto “Calentar” y de 0.25 al conjunto “Nada”.

4.3.1.3. Proceso de *Defuzzificación*

El bloque de reglas proporciona las variables de salida en forma de variables lingüísticas. Por último, las etiquetas lingüísticas y los grados de pertenencia a las mismas deben ser convertidos a un valor numérico de actuación que es la señal que finalmente se manda a los actuadores. Esto se consigue mediante el proceso de *defuzzificación* que se detalla a continuación. Se ha usado el término *defuzzificar* que es un anglicismo adaptado dado que no existe equivalente en español. Su significado vendría a ser "revertir el proceso de hacer difuso y obtener una salida numérica precisa".

Este proceso es similar a invertir el proceso de convertir las señales de entrada en etiquetas. Hay dos formas diferentes de realizar el proceso, llamadas Takagi-Sugeno (Takagi, 1985) y Mamdani (Mamdani, 1975). El proceso Mamdani es el más extendido pero requiere de un mayor coste computacional y la salida no tiene porqué ser continua. Por el contrario el método Takagi-Sugeno es mucho más eficiente, más lineal y la salida es siempre continua.

El proceso Mamdani utiliza diversos métodos de *defuzzificación*. En la figura 4.6 se ejemplifica uno de ellos: *defuzzificación* mediante el centro de

áreas. En este método se calcula la actuación final usando la entrada y las funciones de pertenencia de la 4.3.1 y las reglas usadas como ejemplo en la sección 4.3.1.2. Como puede observarse en la figura 4.6, este método implica calcular el centro de áreas de geometrías arbitrarias que se modifican en cada ciclo de computación. Esta es la razón del mayor coste computacional del método Mamdani. Según este método, en el caso del ejemplo, habría que girar el mando 20 grados en el sentido adecuado para calentar.

Figura 4.6. Proceso de *defuzzificación* usando el método Mamdani: la actuación viene marcada por el centro de áreas de la unión de funciones de pertenencia marcadas por las reglas, recortadas según el grado de pertenencia de las entradas correspondiente.

Por otra parte, el método Takagi-Sugeno en lugar de asignar funciones de pertenencia a las etiquetas de salida, asigna valores numéricos directamente a las variables de salida y calcula la suma lineal. Siguiendo con el ejemplo anterior, una *defuzzificación* tipo Takagi-Sugeno consistiría en:

Calentar es girar -35° .

Nada es girar 0° .

Enfriar es girar 35° .

La actuación final sería:

75% de -35° más 25% de $0^{\circ} \rightarrow -26.25^{\circ}$.

El resultado, como era de esperar, es parecido al obtenido por el otro método. Sin embargo, es evidente la rapidez de este método pues con una combinación lineal de las actuaciones marcadas por las reglas se obtiene la salida final directamente. En esta tesis se ha usado este método en el control difuso, como se explica más adelante.

5. EL PROYECTO QB50 Y EL SATÉLITE QBITO

5.1. Introducción

En este capítulo se va a exponer una introducción al proyecto en el que se engloba el algoritmo de control estudiado en esta tesis. Se va a explicar en qué consiste el proyecto QB50, en el que se engloba el satélite QBITO, así como QBITO en sí mismo. En concreto se va a describir la misión del satélite, con las cargas de pago embarcadas. Además se van a detallar tanto la arquitectura como los subsistemas que componen QBITO para tener una visión global del satélite. Por último se va a describir más en detalle el *hardware* del módulo de actitud (con especial énfasis en los actuadores disponibles) por ser éste de especial interés para esta tesis.

5.2. El proyecto QB50

La termosfera es una región de la atmósfera situada por encima de los 85 km de altura y sin límite superior. La zona inferior, conocida como baja termosfera es una región de difícil estudio debido a que la tenue atmósfera presente provoca fuerzas aerodinámicas sobre los satélites que dificultan el estudio prolongado de la misma. En el pasado se han realizado observaciones mediante los satélites Explorer de la Agencia Espacial Americana (NASA, por sus siglas en inglés: *National Aeronautics and Space Agency*) que seguían órbitas muy elípticas (perigeo de 200 km y apogeo de 3.000 km típicamente). Estos satélites hacían mediciones a su paso por el perigeo (que se encuentra dentro de la termosfera). Sin embargo, el tiempo pasado en esta zona era de tan solo algunas decenas de minutos. Por otro lado, también se han hecho intentos de observación mediante el uso de cohetes de sondeo, siendo actualmente la única fuente de mediciones *in situ*. El mayor inconveniente de este método es que el tiempo disponible para tomar medidas en esta región sigue siendo muy corto (algunos minutos), además solo se pueden hacer mediciones en la parábola que describen por lo que variaciones espacio-temporales pueden pasar inadvertidas. Típicamente, los cohetes de sondeo son una excelente herramienta para bajas altitudes (en el entorno de los cien kilómetros) siendo menos utilizados para alturas mayores.

La termosfera es, por lo tanto, la capa menos explorada de la atmósfera y para intentar solucionarlo se ha puesto en marcha el proyecto QB50 (Von

Karman Institute, [online]). Este proyecto está dirigido por el *Von Karman Institute* y financiado por el 7º Programa Marco europeo que prevé lanzar 50 nanosatélites prácticamente a la vez. Éstos se espera que se distribuyan de manera casi homogénea a lo largo de su órbita (como si de un collar de perlas espaciales se tratase), debido a las diferentes propiedades físicas y aerodinámicas de cada satélite participante. El proyecto QB50 proporcionará medidas *in situ* multipunto durante un periodo extenso, del orden de meses. Para conseguir una completa comprensión de esta fascinante región, la misión cuenta con tres tipos de sensores distribuidos entre los 50 satélites participantes en el proyecto.

La órbita de los satélites ha variado a lo largo del proceso de diseño. En etapas tempranas de diseño, la órbita inicial era heliosíncrona circular de 380 km de altura y aproximadamente 98° de inclinación. Sin embargo, debido a problemas con el lanzador, solo seis de los cincuenta satélites van a seguir una órbita heliosíncrona, siendo la mayoría desplegados desde la Estación Espacial Internacional (ISS por sus siglas en inglés: *International Space Station*). Los satélites participantes van a efectuar mediciones coordinadas en tres órbitas diferentes:

- 6 CubeSats en una órbita heliosíncrona a 475 km, siendo lanzados por el lanzador indio *Polar Satellite Launch Vehicle* (PSLV).
- 40 CubeSats en una órbita similar a la de la ISS, de unos 415 km de altura e inclinación ~50°.
- 2 CubeSats de demostración tecnológica en órbita heliosíncrona a 550 km lanzados en un Dnepr.

5.3. El satélite QBITO

Una propuesta presentada para la misión QB50 por el grupo de investigación del E-USOC (*Spanish User Support and Operations Centre*) (E-USOC, [online]) fue seleccionada y se está llevando a cabo desde el año 2012 (figura 5.1). El satélite desarrollado se llama QBITO y es el primer satélite de formato CubeSat desarrollado por la Universidad Politécnica de Madrid (Barrios, 2014). QBITO es un satélite que sigue el estándar CubeSat y que tiene un factor de forma de 2U (CubeSat, [online]).

QBITO ha sido desplegado en órbita desde la ISS con una órbita de algo más de 400 km de altura y 51.6° de inclinación. Debido a la resistencia

aerodinámica de los satélites, la altura de los mismos disminuirá progresivamente hasta que acaben desintegrados en la atmósfera.

Figura 5.1. El satélite QBITO junto al ingeniero de actitud Daniel Calvo.

5.3.1. La misión del satélite QBITO

Como todo satélite, la piedra angular sobre la que se construye una misión son las cargas de pago. En concreto, QBITO tiene cuatro cargas de pago que se describen a continuación. Como integrante del proyecto QB50, QBITO lleva embarcada como carga de pago principal un espectrómetro de masas (INMS por sus siglas en inglés: *Ion-Neutral Mass Spectrometer*) desarrollado por el *Mullard Space Science Laboratory* y que estudia la volatilidad de la resistencia aerodinámica residual así como la química de la termosfera y el impacto de la meteorología espacial en la atmósfera alta.

Además del INMS, QBITO lleva tres cargas de pago más para maximizar el retorno científico de la misión. Un experimento sobre el uso de n-Octadecano como material de cambio de fase para su uso en control térmico en futuras misiones (PCM por sus siglas en inglés: *Phase Change Material*). Este experimento está desarrollado por la Universidad de Lieja en colaboración con el E-USOC. Un estudio del comportamiento de un detector PbSe de radiación en el infrarrojo medio no refrigerado basado en el método de deposición en fase vaporosa (MWIRD por sus siglas en inglés: *Medium Wave InfraRed Detector*) desarrollado por *New Infrared Technologies*, que es

una empresa especialista en esta tecnología. Por último, un *software* experimental de control de actitud (ESW por sus siglas en inglés: *Experimental Software*) que se probará en vuelo, donde podrá ser comparado con un control alternativo ya validado. Este SW, basado en algoritmos de lógica difusa es la base de esta tesis. En la figura 5.2 puede verse una representación de QBITO con los experimentos y actuadores etiquetados.

Figura 5.2. Vista abierta del satélite QBITO con los experimentos y actuadores (sección 5.3.7) etiquetados.

El INMS (experimento principal de QBITO) impone ciertas restricciones, que afectan directamente a la actitud del satélite, para su correcto funcionamiento. La más restrictiva es que el eje $\mathbf{X}_B$ del satélite apunte hacia el vector velocidad con un error de apuntamiento que no puede ser superior a diez grados (representado como un cono con eje el de la velocidad orbital y con semiángulo (α) de diez grados en la figura 5.3) durante la toma de datos científicos con el INMS. Además, la actitud real del satélite debe conocerse con un error menor que dos grados (representado como un cono de semiángulo (β) de dos grados en la figura 5.3). Ambos requisitos aplican desde el inicio de la misión hasta una altura de 200 km, lo que implica que el control debe de ser capaz de funcionar a pesar de existir perturbaciones significativas. La actitud real debe estar siempre dentro del cono gris claro de la figura 5.3, teniendo en cuenta que el cono oscuro representa el mayor error admisible en la determinación. Los ejes $\mathbf{X}_O$, $\mathbf{Y}_O$ y $\mathbf{Z}_O$ son el sistema de

referencia ORF y los X_B , Y_B y Z_B son los ejes BRF. El vector V representa el vector velocidad orbital.

Figura 5.3. Requisitos impuestos por el INMS para la misión QBITO. Los ejes X_O , Y_O y Z_O representan el sistema de referencia ORF y los X_B , Y_B y Z_B son los ejes BRF.

5.3.2. Subsistemas de QBITO

QBITO se compone de cuatro subsistemas principales además de otros secundarios en los que no se va a entrar en detalle (v. g. el subsistema de control térmico):

- Subsistema de potencia (EPS por sus siglas en inglés: *Electric Power Subsystem*).
- Subsistema del ordenador de a bordo (OBC por sus siglas en inglés: *Onboard Computer*).
- Subsistema de comunicaciones o COM.
- Subsistema de actitud.

Estos subsistemas están interconectados (además de físicamente) a través de las líneas de potencia y de comunicaciones como se detalla en la figura 5.4. Las comunicaciones entre módulos se hacen, principalmente,

mediante el uso de un bus I2C dedicado, en el que el OBC ejerce de maestro. Además existen otros buses I2C secundarios, uno para el EPS y otro para el ADCS, que se usan para la intercomunicación de los distintos módulos que forman cada subsistema. Tanto el OBC como el ADCS disponen de sendas líneas de comunicación serie (UART por sus siglas en inglés: *Universal Asynchronous Receiver-Transmitter*) que permiten la depuración del código embarcado desde el exterior. Por último, existen además otras líneas de comunicación entre subsistemas que no se van a detallar aquí.

Figura 5.4. Esquema de conexiones de QBITO (Barrios, 2013).

En cuanto al *hardware*, el EPS y el COM han sido desarrollados íntegramente por el equipo de QBITO mientras que el ADCS ha sido provisto por el consorcio y el OBC ha sido adquiridos a una empresa externa: *Surrey Space Centre* (SSC por sus siglas en inglés). Para simplificar el proceso de diseño, se ha escogido como placa del OBC un CubeComputer, que es la misma placa que emplea el ADCS.

Se va a describir sucintamente cada uno de los subsistemas principales por separado.

5.3.3. Subsistema de potencia

El EPS (figura 5.5) es el responsable de administrar la distribución de electricidad a bordo. La electricidad se obtiene a través de 18 células solares presentes en cinco de las seis caras del satélite (en la cara frontal se instala el INMS y no permite la instalación de paneles). El EPS proporciona al resto de subsistemas diversas líneas de potencia a tres niveles de voltaje diferente: 3.3V, 5V y el voltaje de baterías. Éste último no está regulado por lo que estará en torno a 8V pero sin valor fijo pues está conectado directamente a las baterías. Dichas baterías son gestionadas también por el EPS, que es el encargado de almacenar en ellas toda la energía sobrante una vez se ha usado la que necesitaba el satélite, así como de proveer potencia cuando los paneles no están iluminados (eclipse) o cuando la potencia que suministran no es suficiente para cubrir el consumo de los subsistemas y cargas de pago.

Figura 5.5. Módulo de baterías y EPS.

5.3.4. Subsistema del ordenador de a bordo

El OBC (figura 5.6) es el cerebro de QBITO: administra todas las tareas que tienen que ejecutarse mientras el satélite esté operativo y gestiona los modos de operación y toda la lógica subyacente. En concreto es el encargado de iniciar los procesos de los experimentos en las ventanas establecidas para ello. Además se encarga de correr los procesos del módulo de COM ya que éste no posee un microcontrolador dedicado, a diferencia del EPS y el ADCS que sí lo tienen. Por lo tanto todas las comunicaciones de QBITO pasan a

través del OBC que luego gestiona y distribuye los comandos y telemetrías según corresponda.

Dado que tanto el ADCS como el EPS o el mismo OBC son críticos para la operación de QBITO, existen procesos de "perro guardián" (*whatchdog* en inglés), tanto internos como externos que garantizan un correcto funcionamiento del satélite completo. Estos procesos reinician el sistema (ADCS, OBC etc.) en caso de que se quede bloqueado más tiempo del permitido. En el caso del ADCS, cada ciclo de control reinicia el contador del *watchdog* y en caso de que pasen más de ocho segundos desde el último reinicio del contador, el *SW* se reinicia inevitablemente.

Figura 5.6. Placa de Cubecomputer. Tanto el OBC como el ADCS usan una placa como esta como cerebro del subsistema.

5.3.5. Módulo de comunicaciones

El módulo de comunicaciones, o COM, se encarga de establecer dos enlaces de datos con las estaciones de tierra, uno de subida y otro de bajada. El enlace de bajada es a 9600 bits por segundo (bps) y se realiza en la banda de ultra alta frecuencia (UHF por sus siglas en inglés: *Ultra High Frequency*) a 436 Mhz mediante una antena formada por dos dipolos en molinete que consiguen polarización circular y un patrón omnidireccional. El enlace de subida funciona a 1200 bps y se realiza mediante un monopolo sencillo en la

banda de muy alta frecuencia (VHF por sus siglas en inglés: *Very High Frequency*) a 135 MHz.

Dentro de COM existen 3 módulos separados: placa de desfase, placa de antenas y placa del transceptor. La primera adapta la señal que se quiere emitir para el correcto funcionamiento de la antena de bajada. En concreto desfasa la señal para cada uno de los dipolos de la antena UHF, de ahí su nombre. La placa de antenas aloja tanto las antenas propiamente dichas como sus mecanismos de despliegue. Por último, la placa del transceptor es la encargada de modular la señal que se desea emitir, así como demodular las señales que se reciben en la antena de VHF. Todas las señales usan modulación en frecuencia para su modulación.

5.3.6. Módulo de control de actitud

Por último, el módulo del ADCS (figura 5.7), que es en el que más se va a profundizar por ser de especial interés para esta tesis ya que es responsable de determinar y controlar el estado dinámico del satélite en todo momento. En las etapas iniciales de diseño se decidió que lo más interesante era centrarse en la parte SW por lo que el *hardware* es un módulo comercial proporcionado el consorcio. Este módulo está compuesto por 3 placas de circuitería integrada (PCB por sus siglas en inglés: *Printed Circuit Board*): CubeComputer, Cubecontrol y CubeSense como se puede observar en la figura 5.7. En el primero de ellos está alojado el microprocesador así como los periféricos necesarios para su funcionamiento (i.e. memorias, buses etc.) y se comunica con los otros dos módulos mediante un bus I2C dedicado.

El módulo de CubeControl interacciona con la mayoría de sensores y actuadores del satélite y tiene a su vez dos microprocesadores redundantes, que proporcionan funcionalidades de control limitadas en caso de un único fallo simple. Por último el módulo CubeSense se encarga de gestionar las dos cámaras (descritas a continuación) de a bordo así como de procesar las imágenes que adquieren, para proporcionar medidas derivadas más útiles para el proceso de determinación de actitud.

El módulo de *hardware* (figura 5.8) del ADCS, está provisto de diversos sensores a disposición de CubeComputer: una de las cámaras del CubeSense actúa como sensor de nadir, la otra cámara (con un filtro específico para observación solar) actúa como sensor de sol, cinco sensores de sol de baja

precisión, un giróscopo de un eje, un magnetómetro de tres ejes y un GPS. Los actuadores presentes se detallan en la sección 5.3.7.

Figura 5.7. *Hardware* del módulo del ADCS. Control, Computer y Sense son las placas del ADCS: CubeControl, CubeComputer y CubeSense respectivamente.

La cámara de nadir apunta según el eje Z_B y puede proporcionar imágenes en blanco y negro de hasta 640 por 480 píxeles. La cámara de sol apunta hacia $-Z_B$ y también permite tomar imágenes de las mismas características. Los sensores de sol de baja precisión se encuentran en todas las caras salvo la ocupada por el INMS. El giróscopo está montado sobre la placa de CubeControl y mide el giro según el eje Y_B . Por último, el magnetómetro está integrado en un mecanismo desplegable, montado sobre la cara Z_B .

Figura 5.8. El ingeniero de sistemas de QBITO sostiene el módulo del ADCS completo.

El SW del ADCS lleva una *suite* completa de determinación y control de actitud bajo la forma del experimento ESW. El núcleo de este SW es un control basado en lógica difusa objeto de esta tesis. Éste interacciona tanto con los periféricos del ADCS (i. e. sensores y actuadores) como con el resto del satélite mediante comunicación con el OBC por I2C. Por último, hace de interfaz para que QBITO pueda disponer de medidas del GPS, tanto para conocer su posición como, mucho más importante, para conocer la hora de manera precisa dado que, debido a la magnitud de la velocidad orbital, un pequeño error en la medida del tiempo provoca un error muy grande en la posición estimada del satélite.

Finalmente, para no poner en riesgo la misión en caso de error en el nuevo algoritmo difuso o en el SW del ADCS en general, un SW ya validado en vuelo, que incluye un control clásico, es embarcado en QBITO como el ADCS primario. Una vez en órbita se probarán ambos algoritmos en escenarios similares para analizar su respuesta, así como la precisión conseguida y el consumo energético. Como el algoritmo primario no estaba disponible para hacer pruebas en tierra en la etapa de diseño, se ha desarrollado un PID específico que se usa como referencia en todas las pruebas que se realizan al control difuso en tierra.

5.3.7. Actuadores de QBITO

Los actuadores presentes en QBITO son de dos tipos: rueda de momento y magnetopares (figura 5.7). Es importante estudiarlos en detalle pues son los que limitan el desempeño de cualquier algoritmo que se use. QBITO dispone de 3 magnetopares (electroimanes), alineados con cada uno de los ejes del sistema de referencia ligado al cuerpo. Los magnetopares de los ejes $\mathbf{Y}_B$ y $\mathbf{Z}_B$ tienen núcleo ferromagnéticos y forma alargada; el correspondiente al eje $\mathbf{X}_B$ esté al aire (no tiene núcleo) y tiene una superficie equivalente al área frontal del QBITO. Al carecer de núcleo, el magnetopar puede ser empaquetado entre las placas CubeSense y CubeComputer. Como actuador mecánico se tiene una rueda de momento cuyo eje es paralelo al eje $\mathbf{Y}_B$, que proporciona control y estabilidad giroscópica en ese eje. Esta configuración se ha elegido ya que como se requiere que el satélite apunte hacia el vector velocidad y este va rotando con la órbita, el satélite deberá girar alrededor de su eje $\mathbf{Y}_B$, alineado con el eje $\mathbf{Y}_O$, a una velocidad de aproximadamente 0.07 °/s. Por ello, el actuador principal será la rueda de momento dado es capaz de proporcionar el par requerido de manera continuada, lo que no se puede asegurar con los magnetopares. Además, los magnetopares necesitan mayor energía para conseguir el mismo par y su actuación depende del campo magnético terrestre y no puede asegurarse que éste vaya a tener siempre la dirección adecuada para los objetivos de la misión. Por lo tanto, la directiva es maximizar el uso de la rueda frente a los magnetopares, reduciendo así la dependencia con el campo magnético terrestre.

6. ELABORACIÓN DE LAS HERRAMIENTAS DE SOPORTE NECESARIAS

Hasta ahora solo se ha estudiado el bloque de control de la figura 3.2, sin embargo, para poder diseñar, ajustar y probar dicho bloque de control es necesario disponer de una planta sobre la que poder actuar. Como los algoritmos que se van a diseñar tienen como objetivo el control de actitud de un satélite en órbita, es imposible usar la planta real en la etapa de diseño. Por ello es imprescindible crear un simulador que represente a la dinámica del satélite en las fases de diseño. Este tipo de simuladores, llamados simuladores DKE (por sus siglas en inglés: *Dynamics, Kinematics and Environment*) van desde unas simples ecuaciones de un sólido rígido en el vacío absoluto hasta simuladores complejos que recogen diferentes elementos perturbadores así como modelos más completos del satélite. Por otro lado, también hay que modelar el comportamiento de los sensores y actuadores, pues la respuesta del control puede verse influenciada por ellos.

6.1. Sensores y actuadores

El simulador debe ser desarrollado en etapas muy tempranas del estudio para poder servir de base al resto de procesos de diseño. Sin embargo, tanto las especificaciones de sensores como las de los actuadores todavía no están bien definidas, salvo gruesas pinceladas. Por lo tanto, se decide que para el bloque de simulación los sensores van a ser obviados y los actuadores linealizados. La figura 6.1 esquematiza cómo se simplifican los bloques de la figura 3.2 teniendo en cuenta cómo se han modelizado los sensores y actuadores. Esto quiere decir que el algoritmo de control conoce el estado dinámico del satélite sin errores, por lo que la etapa de determinación de actitud no es necesaria. Por lo tanto, el bucle de realimentación en la figura 6.1 se hará directamente sin pasar por ningún proceso extra. Por supuesto, el control real no conoce su estado dinámico directamente y éste debe ser estimado por un algoritmo de determinación. Este proceso, inevitablemente, va a introducir errores en el control por lo que éste debe ser lo suficientemente robusto como para no ver empañadas sus prestaciones cuando se pruebe en el satélite en órbita.

Los actuadores han sido modelados, de manera extremadamente simplista, porque es necesario introducir limitaciones en la velocidad y potencia de actuación pero al mismo tiempo no se tenía toda la información

necesaria para un modelo más preciso. De esta manera, el modelo de los actuadores se comporta linealmente con la señal de entrada (proporcionando la actuación requerida exactamente) hasta que se llega al límite de actuación, momento en el que el actuador satura y sigue dando la misma actuación aunque la señal siga aumentando. En términos matemáticos se puede resumir como en la ec. (6.1).

$$\tilde{T} = \begin{cases} T_{\max} & \text{if } T > T_{\max} \\ T & \text{if } T_{\min} < T < T_{\max} \\ T_{\min} & \text{if } T < T_{\min} \end{cases} \quad (6.1)$$

Donde $\tilde{T}$ es la actuación efectiva después de pasar la señal T por el modelo del actuador. Esta señal T corresponde a cada una de las componentes del vector par de control ($\mathbf{T}$). Por supuesto, tanto la actuación en aceleración de la rueda como su velocidad sufren un proceso de saturación equivalente al representado en la ec. (6.1).

Figura 6.1. Esquema simplificado del control.

Como ya se ha mencionado en la sección 5.3.7, QBITO dispone de 3 tipos de actuadores diferentes: magnetopar al aire, magnetopar con núcleo ferromagnético y rueda de momento. Para este estudio los tres tipos de actuadores se han modelizado de la siguiente manera:

- Magnetopar al aire: máximo dipolo (D) 0.12 Am².
- Magnetopar con ferrita: máximo dipolo 0.2 Am².
- Rueda: velocidad máxima 8000 rpm, velocidad mínima 4000 rpm, aceleración máxima 1000 rpm/s.

Las actuaciones máximas aquí descritas son independientes del signo. Todo el estudio expuesto en esta tesis se ha hecho usando los actuadores descritos aquí. Sin embargo, los actuadores y su *firmware* (SW interno del

actuador) han sido modificados por el fabricante cuando tanto los modelos como su análisis ya habían sido completados, por lo que los valores arriba descritos han quedado obsoletos. En concreto, las principales diferencias son que la rueda tiene una aceleración máxima significativamente menor y que los magnetopares, en lugar de poder establecer una fracción del dipolo máximo, actúan siempre con el máximo dipolo pero solo durante una fracción de cada segundo.

Como en el proceso de calibración utilizado (ver sección 7.3) se ha primado reducir el coste sacrificando agilidad, las actuaciones de la rueda siguen siendo válidas pues nunca se supera el nuevo umbral. En el caso de los magnetopares, asegurar que la nueva definición de su modo de operación no altera al resultado del control ya implementado es bastante más complejo dado que cambia la forma de activarse, no la magnitud de la activación.

Antes de avanzar, es interesante remarcar que para simular correctamente la nueva operación de los magnetopares se requieren 1.000 veces más pasos de simulación que con el modo de operación inicial (actuación como un dipolo de magnitud una fracción del dipolo máximo). Aproximadamente esto supone multiplicar por mil el tiempo de simulación (i. e. las simulaciones efectuadas en el capítulo 9 habrían necesitado más de 20 años para completarse).

Para comprobar que el control ya implementado funciona de manera aceptable con los nuevos magnetopares, sin necesidad de repetir todas las simulaciones hechas hasta la fecha, se efectúan una serie de pequeñas simulaciones en las que se compara la respuesta global del satélite con ambos tipos de operación de los magnetopares. En estas simulaciones se observa que la respuesta global es prácticamente idéntica usando ambas aproximaciones (figura 6.2). Por lo tanto se extrapola que el control ya diseñado sigue siendo válido con las nuevas definiciones de los actuadores.

Figura 6.2. Diferencias en actitud tras actuación a dipolo constante en el eje Y_B (0.15 Am^2) usando el modelo original con respecto al uso del nuevo modelo (actuación de 0.2 Am^2 durante el 75% del tiempo).

6.2. Simulador de actitud

El núcleo del DKE son las ecuaciones del movimiento (ec. (6.2)) de un satélite con elementos de intercambio de momento (ruedas de reacción o momento) (Sidi, 2000). En la ec. (6.2) los dos primeros términos del lado derecho de la ecuación describen la dinámica de un sólido rígido sometido a pares externos. Así mismo, los dos últimos términos representan el par equivalente de reacción que se necesita para rotar un segundo sólido (rueda de momento) solidariamente al primero, mientras gira sobre sí mismo.

$$d\mathbf{\Omega} / dt = \mathbf{I}^{-1} \left(\mathbf{T} + \mathbf{I}\mathbf{\Omega} \times \mathbf{\Omega} - \mathbf{I}_w \frac{d\mathbf{\omega}_w}{dt} + \mathbf{I}_w \mathbf{\omega}_w \times \mathbf{\Omega} \right) \quad (6.2)$$

En esta ecuación $\mathbf{\Omega}$ es la velocidad angular del satélite con respecto al sistema de referencia ECI, expresada en ejes BRF. $\mathbf{T}$ son los pares externos del sistema, bien de perturbación (v. g. par de la resistencia aerodinámica) bien actuaciones de control (v. g. la actuación de los magnetopares). $\mathbf{\omega}_w$ es la velocidad de rotación de la rueda de momento en ejes BRF. Por último, $\mathbf{I}$ e $\mathbf{I}_w$

son los tensores de inercia del satélite y de la rueda de momento respectivamente.

La ec. (6.2) determina el ratio de variación de la velocidad angular del satélite con respecto al sistema de coordenadas ECI expresado en ejes BRF. Al integrar a lo largo del tiempo, y con las condiciones iniciales adecuadas obtenemos el valor de la velocidad angular instantánea del satélite. Ésta sirve para obtener la actitud instantánea expresada en cuaterniones (ver sección 3.1.2). Los cuaterniones se obtienen a partir de las siguientes ecuaciones:

$$\mathbf{q}(t) = \int_0^t \dot{\mathbf{q}}(t) dt \quad (6.3)$$

$$\dot{\mathbf{q}}(t) = \frac{1}{2} \mathbf{q}(t-1) [\boldsymbol{\Omega}(t) \quad 0] \quad (6.4)$$

Donde el símbolo $\cdot$ representa el producto de cuaterniones definido como:

$$\begin{aligned} \mathbf{q}_i &= [\vec{\mathbf{q}}_i \quad q_{0i}] \rightarrow \\ \mathbf{q}_1 \cdot \mathbf{q}_2 &= [q_{0_2} \vec{\mathbf{q}}_1 + q_{0_1} \vec{\mathbf{q}}_2 + \vec{\mathbf{q}}_1 \times \vec{\mathbf{q}}_2 \quad q_{0_1} q_{0_2} - \vec{\mathbf{q}}_1 \vec{\mathbf{q}}_2^T] \end{aligned} \quad (6.5)$$

El símbolo $\times$ representa el producto vectorial de dos vectores.

Usando las ec. (6.3) y ec. (6.4) se obtiene la actitud del satélite respecto al sistema de referencia ECI tal y como se detalla en la ec. (6.6) Por simplicidad se ha denotado mediante subíndices el instante de tiempo de cada una de las variables implicadas.

$$\mathbf{q}_t = \frac{1}{2} \int_0^t [q_{0_{t-1}} \boldsymbol{\Omega}_t + \vec{\mathbf{q}}_{t-1} \times \boldsymbol{\Omega}_t \quad -\vec{\mathbf{q}}_{t-1} \boldsymbol{\Omega}_t^T] dt \quad (6.6)$$

Estos cuaterniones representan la actitud del satélite respecto a ejes ECI expresados en ejes BRF. Sin embargo, los requisitos de apuntamiento impuestos por el INMS están expresados respecto a los ejes ORF (figura 5.3) por lo que para realizar un control efectivo se necesita disponer del cuaternión que expresa la actitud del satélite respecto a los ejes ORF expresados en ejes BRF. Esto se calcula fácilmente mediante la composición de cuaterniones:

$$\mathbf{q}_{OB} = \mathbf{q}_{IO} * \mathbf{q}_{IB} \quad (6.7)$$

Donde $*$ es el operador conjugado, $\mathbf{q}_{OB}$ es el cuaternión de la actitud del sistema ORF respecto al sistema BRF, $\mathbf{q}_{IB}$ es el cuaternión de la actitud del satélite respecto al sistema ECI obtenido mediante la ecuación (6.6). Por último, $\mathbf{q}_{IO}$ es la actitud de los ejes ECI respecto a los ejes ORF que se obtiene a partir de la ecuación:

$$\mathbf{R}_{IO} = \begin{bmatrix} \frac{\vec{\mathbf{h}} \times \vec{\mathbf{r}}}{|\vec{\mathbf{h}} \times \vec{\mathbf{r}}|} \\ \vec{\mathbf{h}} \\ \vec{\mathbf{r}} \end{bmatrix} \quad (6.8)$$

Donde $\mathbf{R}_{IO}$ es la actitud de los ejes ECI respecto a los ejes ORF en forma de matriz de cosenos directores (matriz de rotación) que debe ser convertida a cuaterniones para su posterior uso en forma de $\mathbf{q}_{IO}$. $\vec{\mathbf{r}}$ es el vector radio orbital unitario en ejes ECI y $\vec{\mathbf{h}}$ es el vector momento angular orbital unitario, calculado según la ecuación:

$$\vec{\mathbf{h}} = \frac{\vec{\mathbf{v}} \times \vec{\mathbf{r}}}{|\vec{\mathbf{v}} \times \vec{\mathbf{r}}|} \quad (6.9)$$

Donde $\vec{\mathbf{v}}$ es el vector velocidad orbital unitario.

Por último, para convertir $\mathbf{R}_{IO}$ al cuaternión $\mathbf{q}_{IO}^*$ de la ec. (6.7) se pueden utilizar múltiples métodos, dependiendo de si la traza de la matriz de rotación es estrictamente mayor que uno o no. Sin embargo, en la gran mayoría de casos, el cuaternión $\mathbf{q}_{IO}$ se puede calcular de manera sencilla a partir de las siguientes ecuaciones:

$$\begin{aligned} \delta &= \arccos\left(\frac{\text{Tr}(\mathbf{R}_{IO}) - 1}{2}\right) \\ \mathbf{v}^\times &= (\mathbf{R}_{IO} - \mathbf{R}_{IO}^T) \end{aligned} \quad (6.10)$$

Donde $\mathbf{v}^\times$ es la matriz producto vectorial del vector $\mathbf{v}$:

$$\mathbf{v}^\times = \begin{bmatrix} 0 & -v_z & v_y \\ v_z & 0 & -v_x \\ -v_y & v_x & 0 \end{bmatrix} \quad (6.10)$$

De la que es trivial obtener el vector $\mathbf{v}$.

Por supuesto, siempre que se hagan operaciones con cuaterniones, es importante normalizar el cuaternión resultante para asegurarse de que la norma se mantiene siempre lo más cercana al valor unidad, para lo que se procede a su normalización:

$$\mathbf{q} = \frac{\mathbf{q}'}{\sqrt{q_0'^2 + |\vec{\mathbf{q}}'|^2}} \quad (6.11)$$

Donde $\mathbf{q}'$ es el cuaternión antes de normalizarlo. La necesidad de normalización se debe a que el error numérico aparejado al uso de cuaterniones crece como el cuadrado de la norma de dicho cuaternión.

6.2.1. Perturbaciones ambientales

En la ec. (6.2) el término $\mathbf{T}$ engloba todos los pares que se están ejerciendo en cada instante sobre el satélite. Estos pares pueden ser o bien los pares de control ejercidos por los actuadores, o bien pares de perturbación producidos por el ambiente y que, de manera general, intentan sacar al satélite de la actitud deseada.

Como ya se ha visto en la sección 2.2.1, los principales pares de perturbación existentes en una órbita baja son el gravitatorio, el magnético, el aerodinámico y el causado por la presión solar. Estos pares tienen magnitudes muy dispares y dependen de la altura de la órbita, como se observa en la figura 6.3. Por supuesto, las curvas obtenidas en esta ilustración son estimaciones hechas a partir de modelos (tanto ambientales como de las propiedades másicas y físicas del satélite) por lo que los pares reales pueden ser significativamente diferentes a los aquí expuestos.

Figura 6.3. Orden de magnitud de los pares de perturbación en función de la altura de la órbita.

Como se puede observar, entre los 400 y los 200 km, donde se engloba la misión de QBITO, los pares más importantes son el aerodinámico y el magnético causado por el dipolo residual del satélite. Es importante destacar que ambos pares dependen de parámetros y características del satélite y son difícilmente calculables *a priori* por lo que en las etapas de diseño se suelen usar valores típicos.

Las fórmulas que se usan para estimar la magnitud de dichos pares de perturbación son las siguientes (Wertz, 1992):

$$T_{gg} = \frac{3\mu}{2(R_T + H)^3} |I_z - I_y| \sin(2a_1) \quad (6.11)$$

$$T_{sol} = \frac{F_s}{c} (1+r) A \cos(a_2) d \quad (6.12)$$

$$T_{mag} = B \cdot D = 2 \frac{M}{(R_T + H)} D \quad (6.13)$$

$$T_{aero} = \frac{1}{2} \rho C_D A d \quad (6.14)$$

Donde T_{gg} , T_{sol} , T_{mag} , T_{aero} , representan los pares provocados por gradiente gravitatorio, por presión de radiación solar, por dipolo magnético y por resistencia aerodinámica, respectivamente. En las ecuaciones anteriores (de la ec. (6.11) a la ec. (6.14)) aparecen tanto constantes físicas como parámetros que deben ser ajustados a la misión en concreto que se quiera analizar. Antes de detallarlos, es importante remarcar que estas ecuaciones sirven para estimar, de manera simplificada, el orden de magnitud de las mencionadas perturbaciones.

Las constantes físicas que aparecen son μ , R_T , c , M y representan la constante de gravitación para la Tierra, el radio terrestre, la velocidad de la luz y el momento magnético terrestre en los polos, respectivamente. Los parámetros que aparecen son H , ρ , a_1 , a_2 , I_y , I_z , F_s , r , D , B , C_D , A y d y van a ser definidos a continuación. Antes de detallar cada uno de los parámetros, es conveniente recordar, que al ser una estimación inicial, nos interesa quedarnos con los valores más conservadores pues representan el peor escenario posible al que el control deberá enfrentarse.

El parámetro H es la altura de la órbita y variará a lo largo de la misión, siendo su efecto más notable la variación en la densidad atmosférica (ρ). Los parámetros a_1 y a_2 representan ángulos característicos del fenómeno estudiado (ángulo entre el eje $\mathbf{Z}_B$ y la vertical local y ángulo de incidencia de la luz solar con la superficie iluminada respectivamente). Por simplicidad, se escogen de tal manera que maximicen las correspondientes perturbaciones para ser conservadores (valores numéricos de $\frac{\pi}{4}$ y 0 respectivamente).

En cuanto a los momentos de inercia, I_y e I_z , a pesar de que en QBITO son muy parecidos, para la estimación de la figura 6.3 se ha considerado que I_y es la mitad de I_z para que el par de perturbación correspondiente sea proporcional al mayor momento de inercia esperado. F_s es la presión de radiación solar y se establece en 1360 W/m². Además se considera también que el satélite es totalmente especular por lo que el coeficiente de reflexión, r , vale 1. El dipolo residual del satélite, D , se toma como el 1% del dipolo producido por los actuadores (0.02 Am²). B es el campo magnético local, que se estima como una fracción de la intensidad de campo existente en los polos. El coeficiente adimensional de resistencia del satélite (C_D) se estima mediante valores característicos de la bibliografía y adquiere el valor de 2.2. Por último

el área del satélite (A) y el brazo máximo (d) de aplicación de los pares se establecen en 0.03 m^2 y 0.01 m respectivamente.

Como se ha mencionado, las fórmulas anteriores proporcionan unas estimaciones útiles para la etapa de diseño y dimensionado inicial, pero que deben ser refinadas para un uso intensivo. Es por ello que el DKE usa modelos más potentes y calcula de manera precisa y variable en el tiempo muchos de los parámetros que hemos fijado a mano en el párrafo previo. En concreto, respecto a los modelos utilizados, para calcular el par debido a la resistencia aerodinámica utiliza el modelo NRLMSISE-00 (Picone, 2002) para calcular el valor instantáneo de la densidad atmosférica. Este modelo calcula para cada instante la densidad en las coordenadas instantáneas del satélite. Además, como el DKE tiene conocimiento completo del estado dinámico del satélite, puede estimar tanto el área frontal como el centro de presiones en cada instante.

Para la otra perturbación principal, la causada por el alineamiento entre el dipolo residual y campo magnético terrestre, se usa el modelo IGRF-11 (Finlay, 2010) para estimar el campo magnético terrestre en cada instante. En todos los casos el DKE calcula las perturbaciones instantáneas teniendo en cuenta la actitud del satélite para poder diseñar el control en circunstancias lo más parecidas al entorno real de la misión. Por último, el DKE utiliza como propagador orbital un integrador basado en un modelo gravitatorio que incluye el efecto del J_2 . A pesar de que este modelo acumulará errores orbitales significativos para simulaciones largas, estos pueden ser ignorados dado que apenas tienen efectos en la actitud del satélite.

7. DISEÑO DE LOS CONTROLADORES

Como indica el título de la tesis, el objetivo último de la misma es el diseño y análisis de un algoritmo de lógica difusa para el control de actitud de nanosatélites. En concreto, dicho algoritmo se va a usar a bordo del CubeSat QBITO, donde se va a comparar con un algoritmo ya probado en órbita basado en un control de tipo proporcional y derivativo (PD). Sin embargo, dicho algoritmo no está disponible para hacer pruebas en tierra por lo que se ha considerado interesante diseñar un control PID completo para hacer un estudio en tierra y comparar así el controlador basado en lógica difusa con un control clásico como el PID.

7.1. PID

Para evitar singularidades asociadas a la descomposición de giros elegida como representación de actitud, los controles se van a diseñar usando cuaterniones de la forma descrita en la sección 3.1.2. Sin embargo, cuando se estudien los resultados, estos van a ser presentados siempre en su forma de ángulos de Euler en descomposición 321 como se ha explicado en la sección 3.1.1 y se muestra en la figura 3.1

En el diseño del control de tipo PID basado en cuaterniones se han seguido las ecuaciones de (Sidi, 2000) para controles sin término integral. Sin embargo, se ha decidido añadir un término integral coherente con los términos proporcional y derivativo ya presentes, para comparar el control difuso con un control PID, en principio más robusto que un PID. Las variables de control que se van a emplear son:

$$\mathbf{q}_E = \mathbf{q}_S^{-1} \mathbf{q}_T \quad (7.1)$$

$$\boldsymbol{\Omega}_{OB} = \left\{ \Omega_{OBx} \quad \Omega_{OBy} \quad \Omega_{OBz} \right\} \quad (7.2)$$

En la ec. (7.1) los términos $\mathbf{q}_E$, $\mathbf{q}_S$ y $\mathbf{q}_T$ son, respectivamente, el cuaternión del error, el del satélite y el del objetivo, todos ellos expresados en el sistema de referencia ORF. En la ec. (7.2) $\boldsymbol{\Omega}_{OB}$ representa el vector velocidad angular con respecto al sistema ORF en ejes BRF. De aquí en adelante, y salvo cuando se especifique explícitamente lo contrario, todos los vectores estarán referidos al sistema BRF.

El control PID se puede escribir de manera genérica como se observa en la ec. (7.3).

$$\begin{bmatrix} T_{cx} \\ T_{cy} \\ T_{cz} \end{bmatrix} = \begin{bmatrix} 2K_{xp}\mathbf{q}_E(1)\mathbf{q}_E(4) + K_{xd}\Omega_{OBx} + K_{xi}\int\mathbf{q}_E(1)\mathbf{q}_E(4)dt \\ 2K_{yp}\mathbf{q}_E(2)\mathbf{q}_E(4) + K_{yd}\Omega_{OBy} + K_{yi}\int\mathbf{q}_E(2)\mathbf{q}_E(4)dt \\ 2K_{zp}\mathbf{q}_E(3)\mathbf{q}_E(4) + K_{zd}\Omega_{OBz} + K_{zi}\int\mathbf{q}_E(3)\mathbf{q}_E(4)dt \end{bmatrix} \quad (7.3)$$

Donde T_{cj} representa el par de control en el eje j BRF y K_{jp} , K_{jd} y K_{ji} son las ganancias proporcional, derivativa e integral para ese mismo eje j . Esto proporciona 9 ganancias que pueden ser ajustadas para conseguir el control deseado. Este proceso se explica en detalle en la sección 7.3. Observando la ec. (7.3) se pueden definir unos "ángulos de error equivalentes" que simplifican la notación:

$$\begin{aligned} \gamma &= \mathbf{q}_E(1)\mathbf{q}_E(4) \\ \sigma &= \mathbf{q}_E(2)\mathbf{q}_E(4) \\ \mu &= \mathbf{q}_E(3)\mathbf{q}_E(4) \end{aligned} \quad (7.4)$$

Por lo que la ec. (7.3) puede reescribirse usando los "ángulos equivalentes" de la ec. (7.4) como:

$$\begin{bmatrix} T_{cx} \\ T_{cy} \\ T_{cz} \end{bmatrix} = \begin{bmatrix} 2K_{xp}\gamma + K_{xd}\Omega_{OBx} + K_{xi}\int\gamma dt \\ 2K_{yp}\sigma + K_{yd}\Omega_{OBy} + K_{yi}\int\sigma dt \\ 2K_{zp}\mu + K_{zd}\Omega_{OBz} + K_{zi}\int\mu dt \end{bmatrix} \quad (7.5)$$

Se observa que el control propuesto, tiene tres términos en cada uno de los ejes (por ejemplo para el eje $\mathbf{X}_B$): el término proporcional ($2K_{xp}\gamma$), el término derivativo ($K_{xd}\Omega_{OBx}$) y el término integral ($K_{xi}\int\gamma dt$). El término proporcional actúa con una actuación mayor cuanto mayor sea el error. El término derivativo intenta frenar siempre al satélite, por lo que cuanto más rápido vaya más intentará frenarlo. Por último el término integral intenta que el área de la curva de error en la parte positiva sea igual a la negativa. Esto tiene dos efectos, el primero es provocar que exista siempre *overshoot* (el control siempre se va a "pasar de la raya" antes de llegar a la solución), y el segundo, y más interesante, es que errores pequeños alrededor de la solución tienden a desaparecer con el tiempo. Si no existiese el término integral, la solución estacionaria (para el eje $\mathbf{X}_B$) es aquella que cumple la ecuación:

$$\gamma = \frac{T_{dx}}{2K_{xp}} \quad (7.5)$$

Donde T_{dx} es el par de perturbación en el eje $\mathbf{X_B}$. Esto es así porque se puede considerar que en la solución estacionaria, el par de actuación es idéntico y opuesto al par de perturbación. Sin embargo, por muy pequeño que sea ese error estacionario, al integrarlo a lo largo de tiempos grandes provoca actuaciones significativas que tienden a anular dicho error estacionario.

Como ya se ha explicado, se desea que las actuaciones del eje $\mathbf{Y_B}$ sean llevadas a cabo, principalmente, por la rueda de momento en lugar de por el magnetopar correspondiente. Para convertir la actuación en par a una actuación en aceleración angular de la rueda se usa la siguiente ecuación:

$$\Delta\omega_w = \frac{T_{cy}}{I_w} \quad (7.6)$$

En la ec. (7.6) se ha tenido en cuenta que el tensor de inercia $\mathbf{I_w}$ es diagonal e I_w es el momento de inercia respecto al eje de la rueda. Por otra parte el vector $\boldsymbol{\omega_w}$ tiene un solo elemento no nulo, ω_w . Por tanto, en la ec. (7.6) $\Delta\omega_w$ es la variación de velocidad de rotación de la rueda respecto a su eje.

Con esto queda definido el control PID a falta de calibrar las ganancias de la ec. (7.5), lo que explica en la sección 7.3.

7.2. Lógica difusa

El control por lógica difusa se ha basado en las pautas marcadas en (Jantzen, 2013), en las que se parte de un control de tipo PD clásico para convertirlo en un control puramente difuso. En la figura 7.1 se observa la similitud con un control de tipo PD donde las variables de control son el error y la derivada del mismo.

Dado que la dinámica de cada uno de los ejes es diferente (v. g. las perturbaciones ambientales son prácticamente indiferentes al giro alrededor del eje $\mathbf{X_B}$) y los actuadores disponibles no son iguales en todos los ejes (i. e. el eje $\mathbf{Y_B}$ actuará preferentemente con la rueda de momento y los demás con magnetopares), se diseña un controlador independiente para cada uno de los ejes de igual modo a como se ha procedido en el PID. La figura 7.1 muestra el esquema que se sigue para el controlador difuso. Las variables de control son

las mismas que las utilizadas en la ec. (7.5). Aunque como salida del eje Y también aparece Tc_y , éste solo se usa para desaturar la rueda cuando la velocidad de ésta se acerca a los límites de operación de la misma. Este proceso de desaturación se lleva a cabo utilizando un control difuso diseñado al efecto. Este control está brevemente descrito en la sección 7.2.1 pero no se va a estudiar en profundidad en esta sección porque solo se utiliza en el SW de vuelo, no en las simulaciones. Por lo tanto, para realizar la comparación en tierra entre ambos controladores se ha decidido no incorporarlo al ser algo superfluo en las simulaciones. Las diez ganancias presentes en el esquema se calibran junto con las nueve del PID en la sección 7.3.

Figura 7.1. Esquema del controlador difuso. Los bloques llamados G son las ganancias que hay que calibrar.

Para diseñar un controlador difuso, como ya se ha explicado en la sección 4.3, nos basta con definir las funciones de pertenencia así como las reglas que se aplican a las etiquetas difusas. Antes de presentar las funciones de pertenencia escogidas, hay que matizar algunas consideraciones que se han tenido en cuenta en su diseño.

Dada la geometría de QBITO así como las especificaciones de la misión, y para generalizar al máximo los resultados obtenidos, las funciones de pertenencia deben ser simétricas respecto a la posición central. Esto se debe a que un error negativo es, conceptualmente, igual a uno positivo de la misma magnitud. Además, como los actuadores pueden funcionar exactamente igual en ambos sentidos, todo el control es simétrico respecto al cero. De entre todas las formas posibles que pueden adoptar, solo se van a seleccionar funciones de pertenencia triangulares o, en los extremos, trapezoidales. Esto tiene dos principales razones:

- El coste computacional se reduce por su simplicidad.
- La respuesta es más sensible a pequeñas variaciones alrededor de su centro que la obtenida en funciones de pertenencia con derivada nula en el centro (v. g. curvas Gaussianas).

Como ya se ha dicho, el control debe ser simétrico respecto al cero ya que un error negativo debe provocar la misma respuesta (de signo opuesto) que un error positivo. Por lo tanto, por cada función de pertenencia en la región positiva deberá haber una equivalente en la región negativa. Estas funciones simétricas se establecen alrededor de una función central, dotando al sistema de respuestas diferentes según el punto de operación. Las funciones laterales están ligeramente deformadas hacia el centro, provocando una respuesta más ágil cerca del centro que lejos del mismo. Por otra parte, para conseguir transiciones suaves entre funciones de pertenencia, el corte entre dos funciones consecutivas se produce siempre al 50%. Por último, tomando en consideración todas estas pautas así como el conocimiento adquirido del sistema, las funciones han sido iteradas para mejorar el comportamiento del controlador. De la figura 7.2 a la figura 7.6 se muestran las funciones de pertenencia que han sido finalmente implementadas en el controlador, tanto en las simulaciones como en el SW embarcado, ya que ambos son esencialmente idénticos. La tabla 7.1 describe las abreviaturas empleadas en las gráficas para las variables lingüísticas.

Abreviatura	Variable lingüística
MN	Muy negativo
N	Negativo
PN	Poco Negativo
OK	Correcto
PP	Poco positivo
P	Positivo
MP	Muy positivo

Tabla 7.1. Tabla de abreviaturas que aparecen en las funciones de pertenencia y en la tabla de reglas del control difuso.

Figura 7.2. Reglas de pertenencia difusas para el ángulo σ .

Figura 7.3. Reglas de pertenencia difusas para los ángulos γ y μ .

Figura 7.4. Detalle de la zona central de la figura 7.3.

Figura 7.5. Funciones de pertenencia difusas para la velocidad angular orbital alrededor del eje Y_B (Ω_{OBy}).

Figura 7.6. Funciones de pertenencia difusas para las velocidades angulares orbitales en los ejes X_B y Z_B (Ω_{OBx} y Ω_{OBz}).

En todos los casos, el método de “*defuzzificación*” seguido ha sido el de Takagi-Sugeno (Takagi, 1985). En el método Takagi-Sugeno la salida no tiene funciones de pertenencia sino que se usan funciones lineales que dan directamente un valor de salida numérico, por lo que el proceso de “*defuzzificación*” se simplifica notablemente. En el control aquí descrito las funciones lineales han sido reducidas a valores discretos, representando estos la fracción de actuación posible que se va a ejecutar. Por lo tanto, solo existen funciones de pertenencia para las entradas, usando para la salida una lista discreta de valores que representan la fracción de actuación que se va a usar, bien sea Tc_j o $\Delta\omega_w$. La tabla 7.2 y la tabla 7.3 muestran las reglas que se aplican para pasar de las variables lingüísticas a las salidas numéricas.

		σ				
		MN	N	OK	P	MP
Ω_{OBy}	MN	1	1	1	0.5	-1
	N	1	1	0.5	0	-1
	OK	1	0.5	0	-0.5	-1
	P	1	0	-0.5	-1	-1
	MP	1	-0.5	-1	-1	-1

Tabla 7.2. Reglas difusas para el eje Y_B . Los valores numéricos representan la fracción de actuación con respecto al valor máximo.

		$\gamma \text{ o } \mu$						
		MN	N	PN	OK	PP	P	MP
Ω_{OBx} o Ω_{OBz}	MN	1	1	1	1	0.5	0	0
	N	1	1	1	0.5	0	-0.5	-0.5
	OK	1	0.5	0.5	0	-0.5	-0.5	-1
	P	0.5	0.5	0	-0.5	-1	-1	-1
	MP	0	0	-0.5	-1	-1	-1	-1

Tabla 7.3. Reglas difusas para los ejes X_B y Z_B . Los valores numéricos representan la fracción de actuación con respecto al valor máximo.

Dado que cada controlador tiene dos entradas y una salida (obviando el control del par en el eje Y que tiene un control dedicado y que se detalla en la sección 7.2.1), el conjunto de cada controlador difuso puede representarse como una superficie bidimensional en un espacio tridimensional. Como ejemplo, en la figura 7.7 puede observarse dicha representación para el control en el eje Y_B .

Figura 7.7. Representación del controlador difuso usado en el eje Y_B .

Por último, y al igual que con el control PID, el siguiente paso es el proceso de calibración de las ganancias de ambos controles que se detalla en la sección 7.3.

7.2.1. Desaturación de la rueda de momento

Como ya se ha mencionado, el control en el eje Y_B tiene como actuador primario la rueda de momento, usando la actuación magnética en ese eje solo para desaturar la rueda cuando ésta sobrepase ciertos umbrales. Esta labor recae en un control difuso dedicado exclusivamente a la desaturación de la rueda. Este control se detalla en una sección independiente porque se ha implementado solo para ser usado en órbita. Como ya se ha mencionado, el control difuso utilizado puede describirse usando sus funciones de pertenencia así como las reglas que les aplican. En la figura 7.8 se han representado las funciones de pertenencia para los parámetros de entrada del control: la velocidad de rotación de la rueda y una variable de control que indica si la rueda está saturada o no.

Figura 7.8. Funciones de pertenencia para la desaturación de la rueda.

Las variables de salida son el par de actuación (como fracción del par máximo) y la variable de control que indica si la rueda está saturada o no (S). El valor en el anterior ciclo de computación de esta última variable es la que

entra también como parámetro de entrada. Esta variable vale cero cuando la no está activo el proceso de desaturación y vale uno cuando este proceso está activo. Esta variable sirve para que el control mantenga el proceso de desaturación hasta que la rueda se encuentre en valores cercanos al nominal.

Siguiendo con la notación que se ha usado en la sección anterior, la tabla 7.4 recoge las reglas que ligán las entradas con la salida en par de este control. Por otro lado, la tabla 7.5 recoge las reglas correspondientes para la variable de control S .

		Velocidad rueda				
		Muy lento	Lento	Nominal	Rápido	Muy rápido
S	No	1	0	0	0	-1
	Si	1	1	0	-1	-1

Tabla 7.4. Reglas difusas para el control de desaturación de la rueda. En cada celda se representan los valores de actuación como fracción del par máximo.

		Velocidad rueda				
		Muy lento	Lento	Nominal	Rápido	Muy rápido
S	No	1	0	0	0	1
	Si	1	1	0	1	1

Tabla 7.5. Reglas difusas para el control de desaturación de la rueda. En cada celda se representan los que toma la variable S .

7.3. Calibración

Los dos controladores, el PID y el difuso, tienen ganancias indeterminadas que permiten fijar su punto de operación. A pesar de que el sistema tiene efectos de acoplamiento significativos entre ejes, la tarea de calibrar el control de manera acoplada sería extremadamente costoso por lo que se procede a calibrar el control de cada eje por separado. Posteriormente, en un proceso iterativo, se verifica que la calibración desacoplada es aceptable para el sistema completo. La primera calibración se hizo utilizando el método manual Ziegler-Nichols (Ziegler, 1993). En este método se van incrementando las ganancias de manera manual hasta que se consiguen respuestas determinadas (v. g. oscilaciones estables). Este método suele dar buenos

resultados como primera aproximación en una gran cantidad de problemas, sin embargo, la respuesta obtenida se considera demasiado lenta para los objetivos de la misión, por lo que se ha ideado un método *ad hoc* de calibración para esta misión.

El método de calibración consiste en establecer todas las ganancias, simular una maniobra determinada en cada eje por separado y analizar la respuesta para mejorar las ganancias en la siguiente iteración. El proceso trata de localizar el mínimo de la hypersuperficie del espacio de ganancias, generando un nuevo mallado alrededor de ese mínimo y de un ancho total de, al menos, un paso de la anterior iteración, entendiendo paso como la separación entre puntos adyacentes en el espacio de ganancias. La maniobra elegida es un escalón de cincuenta grados en el eje analizado, dejando el resto en la posición nominal. Esta maniobra se ha escogido de manera arbitraria para permitir una cierta maniobrabilidad pero a su vez, limitar el tiempo de simulación.

También se escoge arbitrariamente el umbral de tiempo permitido para la maniobra (200 segundos), siendo lo suficientemente grande como para que ambos controladores puedan conseguirlo con cierto margen. Como referencia, el mínimo tiempo teórico para esta maniobra, que consistiría en dar la aceleración máxima la mitad del tiempo y el frenado máximo la otra mitad del tiempo, es del orden de 50 segundos para el eje Y_B o el Z_B (algo menos para el eje X_B), por lo que 200 segundos se considera un intervalo de tiempo adecuado para alcanzar el objetivo. Todas las simulaciones se efectúan en una órbita inicial a 380 km de altura. El proceso puede ser esquematizado como se detalla en la tabla 7.6.

Paso	Descripción
1	Establece un juego de ganancias.
2	Parte de una actitud nominal estable.
3	Comanda un objetivo de +50 grados en el ángulo de balanceo y el resto a cero.
4	Graba la actitud y las actuaciones comandadas durante 200 segundos.
5	Repetir los pasos 2-4 comandando cabeceo a +50 grados y el resto a cero.
6	Repetir los pasos 2-4 comandando guiñada a +50 grados y el resto a cero.
7	Repetir los pasos 1-6 para un juego diferente de ganancias.

Tabla 7.6. Secuencia de excitación y simulación del proceso automatizado de calibración.

De entre todos los parámetros que se graban durante las simulaciones, el más importante para el proceso de calibración es la actitud instantánea del satélite. Con ella se calcula el error instantáneo. El error instantáneo utilizado es el valor absoluto de la diferencia entre el ángulo comandado y el objetivo establecido por la calibración, ignorando el estado de los otros dos ángulos que son comandados a cero. Esto se hace para ignorar efectos de acoplamiento que pueden existir. Expresado en términos matemáticos esta definición de error queda así:

$$e = |\vartheta_{spacecraft} - \vartheta_{target}| \quad (7.7)$$

Donde ϑ representa el ángulo comandado en términos de los ángulos de Euler (balanceo, cabeceo y guiñada). Para ello hay que calcular los ángulos de Euler a partir de los cuaterniones mediante las siguientes expresiones:

$$\begin{aligned} \text{balanceo} &= \text{atan} \left(\frac{2(\mathbf{q}_s(2)\mathbf{q}_s(3) + \mathbf{q}_s(1)\mathbf{q}_s(4))}{1 - 2(\mathbf{q}_s(1)^2 + \mathbf{q}_s(2)^2)} \right) \\ \text{cabeceo} &= \text{asin} (2(\mathbf{q}_s(4)\mathbf{q}_s(2) - \mathbf{q}_s(3)\mathbf{q}_s(1))) \\ \text{guiñada} &= \text{atan} \left(\frac{2(\mathbf{q}_s(2)\mathbf{q}_s(1) + \mathbf{q}_s(3)\mathbf{q}_s(4))}{1 - 2(\mathbf{q}_s(3)^2 + \mathbf{q}_s(2)^2)} \right) \end{aligned} \quad (7.8)$$

Finalmente, partiendo del error definido en la ec. (7.7), se calculan los errores promedio (a partir de ahora error residual) así como la desviación típica del error en los últimos cincuenta de los doscientos segundos de la simulación, aplicando los criterios de control siguientes:

$$\bar{e} = \left(\sum_{t=150s}^{t=200s} \frac{|e|}{N} \right) \leq 0.01 \text{ rad} \approx 0.57 \text{ deg} \quad (7.9)$$

$$\sigma_e = \left[\left(\sum_{t=150s}^{t=200s} \frac{(e - \bar{e})^2}{(N-1)} \right) \right]^{\frac{1}{2}} \leq 0.002 \text{ rad} \approx 0.11 \text{ deg} \quad (7.10)$$

Estas condiciones aseguran que la respuesta conseguida está en el entorno de la solución deseada y que la oscilación máxima que se tiene está acotada a un valor menor todavía que la propia solución deseada, por lo que se espera que la solución sea estable y tenga buena precisión.

De todo el juego de ganancias probadas y cuya respuesta cumple los criterios de las ec. (7.9) y ec. (7.10), el juego óptimo se considera aquel que tiene menor coste acumulado. El coste acumulado es la integral de la función de costes definida en la ec. (7.11), que se calcula usando los valores absolutos de todas las actuaciones comandadas (aceleración de la rueda y par magnético) y dos constantes físicas.

$$C(t) = K_w |\Delta\omega_w| + K_T \sum |T_{cj}| \quad (7.11)$$

Las constantes K_w y K_T tienen en cuenta el *hardware* utilizado así como el entorno de la maniobra. Cuando se han ajustado correctamente sus valores la función de coste es una estimación de cuanta energía se ha usado en la actuación. K_w es la potencia necesaria para cambiar la velocidad de la rueda, lo que se puede aproximar como el consumo máximo de la rueda dividido por la máxima aceleración posible de la rueda. K_T es el inverso de un valor del campo magnético de referencia. Estas constantes se han tomado así dado que las especificaciones de los actuadores no estaban disponibles en su totalidad en el momento de hacer el estudio. Además, como ya se ha indicado, los actuadores se simplificaron en el simulador por la misma razón. En este caso particular, las dos constantes se establecen en $K_w \sim 4.9 \cdot 10^{-3} \text{W/(rpm/s)}$ y $K_T \sim 3.3 \cdot 10^4 \text{W/Nm}$.

Una vez realizado de manera iterativa todo el procedimiento se alcanzan las siguientes ganancias óptimas desde el punto de vista del proceso de calibración (tabla 7.7).

		K_p	K_d	K_i
PID	X	5E-5	-1.8E-3	0
	Y	2E-4	-4.1E-3	1.5E-6
	Z	1.8E-4	-4.0E-3	0

		K	$K_{\Omega OB}$	K_{T_c}	K_ω
difuso	γ	34.38	26.18	4.2E-6	-
	σ	5.73	12.74	9E-6	-0.44
	μ	52.14	43.63	9.9E-6	-

Tabla 7.7. Valores obtenidos para las ganancias de ambos controles mediante el proceso automatizado de calibración.

Es importante recalcar que en el control PID que se ha diseñado para la comparación mediante simulaciones, las ganancias integrales para los ejes $\mathbf{X_B}$

y $\mathbf{Z_B}$ son nulas a pesar de haberse dejado libertad para establecer cualquier valor.

8. ANÁLISIS Y COMPARACIÓN

Durante el proceso de calibración, ambos controles han sido optimizados para la misión QBITO, en concreto para el inicio de su misión que es donde más tiempo pasará el satélite, proporcionalmente al tiempo completo de la misión. Sin embargo, en este capítulo ambos controles estudian globalmente para el conjunto de la misión. Esta verificación se lleva a cabo en dos etapas bien diferenciadas. En la primera etapa se comparan ambos controles en escenarios predefinidos y esperables durante la misión real de QBITO. En la segunda etapa, en el capítulo 9 se efectúa un estudio mediante el método Monte Carlo para englobar un amplio espectro de situaciones. En ambas etapas existen dos objetivos principales, siendo el primero la validación del control difuso para su operación en órbita y el segundo su comparación mediante simulaciones con un algoritmo clásico de tipo PID.

8.1. Verificación y análisis.

8.1.1. Test 1

La primera verificación que debe hacerse es que ambos controles son estables manteniendo la actitud nominal a largo plazo. Este test es interesante dado que la carga de pago principal de QBITO tiene como requisito mantener dicha actitud durante largos periodos de tiempo.

Por lo tanto el primer test realizado consiste en simular el comportamiento a lo largo de 10 órbitas manteniendo la actitud nominal (el eje $\mathbf{X}_B$ del satélite tan cercano como sea posible al vector velocidad orbital $\mathbf{V}$). Para analizar los resultados se usa la misma función de coste de la ec. (7.11) pero se emplea una definición de error diferente a la de la ec. (7.7):

$$e_{3D} = 2\text{acos}(\mathbf{q}_E(4)) \quad (8.1)$$

Esto se hace para contemplar en una única variable unidimensional el error presente en los tres ejes. Conceptualmente la ec. (8.1) es la versión tridimensional del error unidimensional de la ec. (7.7). Este error representa la distancia angular entre los dos cuaterniones siguiendo el camino de mínima distancia. Este concepto se puede esquematizar como se observa en la figura 8.1 donde se ha representado el ángulo de error equivalente como la distancia mínima entre dos vectores.

Figura 8.1. Representación del equivalente del error tridimensional (ec. (8.1)) entre dos vectores en lugar de cuaterniones.

Tras efectuar las simulaciones se comprueba que ambos controles mantuvieron el error por debajo de 10^{-3} radianes. El coste de ambos controles fue virtualmente idéntico. Dado que la representación de los resultados no arroja más información que la arriba expuesta, se considera innecesario reflejarlos en forma de gráfica.

8.1.2. Test 2

Tanto la calibración como el Test 1 han sido realizados en la órbita de inyección. Por lo tanto, parece interesante analizar el desempeño de ambos controles cuando hay resistencia aerodinámica que reduzca progresivamente la órbita, pudiendo analizar la misión en su conjunto. En este test se estudia desde la órbita de inyección hasta la altura en la que los controladores no pueden mantener la actitud nominal. Sin embargo, el simulador no está ideado para representar el decaimiento orbital. Por lo tanto se decide hacer muchas simulaciones a diferentes alturas, concentradas principalmente por

debajo de los 200 km. En cada simulación el tiempo estudiado es de una órbita (5400 s).

Los resultados muestran que ambos controles son capaces de mantener la actitud nominal con errores por debajo de una décima de grado para alturas de órbita incluso inferiores a 200 km como se observa en la figura 8.2. Sin embargo, es importante resaltar que la misión acaba en los 200 km de altitud. Además, por debajo de esa cota se puede considerar que empieza el proceso de reentrada por lo que la física del problema sería diferente y los resultados no deberían ser tenidos en cuenta a tan baja altura.

Figura 8.2. Error medio a lo largo de una órbita en relación a la altura de la órbita.

8.1.3. Test 3

Hasta ahora, en el Test 1 y el Test 2, se ha estudiado solo el caso particular en el que se mantiene la actitud nominal. Esta actitud es la de mínima perturbación por lo que no es interesante para analizar los algoritmos de control. No permite que se aprecien las verdaderas diferencias que surgen cuando se estresa el sistema.

El proceso de calibración (sección 7.3) se hace teniendo en mente una maniobra genérica de tipo. En este test se va a estudiar el desempeño en esa misma maniobra a lo largo de toda la misión, (desde altitudes de órbita de 380 km hasta 200 km cada 20 km). Por lo tanto este test se compone a su vez de múltiples subtest de una única maniobra cada uno: a cada altitud de órbita, ambos controladores son comandados a una actitud de cincuenta grados en cada uno de los ejes por separado (siguiendo el mismo procedimiento que el utilizado para la calibración de la tabla 7.6). En los siguientes subtest, una maniobra en {balanceo/cabeceo/guñada} representa que el objetivo se ha impuesto en ese ángulo y se han dejado los demás comandados a cero. Por lo tanto este test completo se compone de treinta subtest para cada controlador por separado: en cada una de las diez alturas de órbita estudiadas, un subtest para cada uno de los ángulos.

Figura 8.3. Respuestas superpuestas de ambos controles a excitación en cada uno de los ángulos por separado. Cada línea representa una simulación diferente, y todas ellas se han hecho a 380 km de altitud.

A pesar de que los subtest se simulan por separado, en la figura 8.3 se muestran superpuestas las respuestas de todos los ángulos (excitados por separado) en la altura de órbita más alta. En la Figura 8.4 se observa la respuesta de ambos controladores en el mismo test realizado a baja altura.

Figura 8.4. Respuestas superpuestas de ambos controladores a baja altura (200 km).

Se puede observar como el control difuso (líneas sólidas) se comporta de manera más lineal, principalmente en balanceo y guiñada, que el PID equivalente (líneas punteadas). En la figura 8.5 se han representado las aceleraciones angulares sufridas en los ángulos de balanceo y guiñada (por ser estos los que presentan un comportamiento más lineal en el control difuso) para ambos controles. Se observa que el control difuso actúa principalmente solo al principio y al final de la maniobra, dejando que el satélite gire libre en la etapa intermedia. En contrapartida, el PID actúa mucho al principio y luego tiene que compensar más para frenar esa misma actuación. Por eso la respuesta del PID en la figura 8.3 llega mucho antes a un entorno de la solución que el difuso, aunque luego le cuesta mucho, comparativamente, corregir ese último error. Se observa que casi todas las curvas tienden a converger muy cerca del entorno del segundo doscientos cincuenta que es cuando se empieza a medir el error residual, ec. (7.9). Esto se explica porque al minimizar el coste, se está primando que el control sea lo más lento posible, pero convergiendo a partir de un cierto instante. Por lo tanto la respuesta tiende a converger solo cuando se mide y no antes, pues eso aparejaría un incremento de coste.

Figura 8.5. Aceleraciones sufridas por el satélite en las maniobras de balanceo y guiñada con ambos controladores.

Si se analiza el test al completo, a lo largo de toda la misión, es interesante centrarse en dos variables de control, el error residual y el coste acumulado. El error que se estudia se calcula usando la definición de error dada por la ec. (8.1) y promediando su valor absoluto en los últimos cincuenta segundos como se hace en la ec. (7.9). Dicho error residual a lo largo de toda la misión (desde inyección hasta reentrada) se puede observar en figura 8.6. En ésta se aprecia que el control PID tiene un error significativo (superior al límite impuesto en la calibración) cuando se comanda una actuación en dos de los ejes. Esto se debe a efectos de acoplamiento entre ejes, como ya se prevé en la sección 7.3, y que se puede observar en la figura 8.7. En el control de cabeceo, sin embargo, el error residual es muy bajo, debido principalmente a su ganancia integral no nula y a la ausencia de acoplamientos.

Figura 8.6. Error residual tras una maniobra de cincuenta grados en distintas alturas a lo largo de toda la misión. La línea negra horizontal punteada representa el límite impuesto en la calibración.

En la figura 8.7 se observa que, al comandar una actuación en guiñada ésta tiene un error residual muy pequeño pero el error en balanceo se incrementa considerablemente. Un efecto similar se produce al comandar una actuación en balanceo. Esto se debe a que durante la calibración se usa una definición de error unidimensional (ec. (7.7)) buscando ignorar estos efectos. Parece evidente que, en este escenario, incrementar las ganancias que controlan el ángulo de guiñada no va a mejorar el desempeño global pues el error se halla en el ángulo de balanceo. Como estos efectos de acoplamiento significativo solo aparecen para el control PID no se considera interesante intentar contrarrestarlo dado que no se va a hacer un uso real de dicho algoritmo. En todo caso, esto otorgaría una cierta ventaja al PID, dado que al no cumplir con los requisitos del error, el control ha tenido que efectuar menos actuaciones, y por tanto el coste real debería ser ligeramente mayor.

Figura 8.7. Respuesta del PID al comando en guiñada. Se observa como el ángulo de balanceo adquiere un nuevo punto de equilibrio distinto del valor cero.

Analizando los resultados para el control difuso se observa que a la altura de calibración los errores están dentro del límite establecido, creciendo exponencialmente (incluso llegando a valores fuera del límite impuesto en la calibración) según se reduce la altura de la órbita. Esto es coherente con una perturbación principal (par producido por la resistencia aerodinámica) de forma exponencial. Además, el error en giro en el eje $\mathbf{X}_B$ permanece remarcablemente bajo dado que en ese eje las perturbaciones son prácticamente inexistentes e independientes de la altura.

De momento solo se ha estudiado la precisión de ambos algoritmos, pero otro factor decisivo (siendo además la variable que se ha minimizado en el proceso de calibración) es su coste o eficiencia al ejecutar dichas maniobras. El coste acumulado se calcula integrando la función definida en la ec. (7.11). En concreto se va a estudiar la variable C calculada según la siguiente ecuación:

$$\tilde{C} = \frac{C_{difuso}}{C_{PID}} \quad (8.2)$$

En la figura 8.8 puede verse la evolución de ese ratio de costes en función de la altura de la órbita simulada. Se comprueba como en las condiciones a las que se han optimizado ambos controles (órbita de inyección) el control difuso es entre un 60% y un 70% más económico que el PID de referencia, o lo que es lo mismo, éste es hasta tres veces menos eficiente que el difuso. Para órbitas bajas, la eficiencia del control difuso disminuye con la altura, quedándose en reducciones de coste de entre 20% y 30%. Sin embargo, aunque el control difuso sea más eficiente en todo el escenario de la misión, la precisión del mismo (el error residual) por debajo de los doscientos cincuenta kilómetros no es considerada suficientemente buena. Por lo tanto, es necesario mejorar el algoritmo difuso hasta que cumpla con el requisito impuesto en todo el escenario de la misión. Por ello se va a repetir el proceso de calibración para intentar conseguir unas ganancias que cumplan con los objetivos de la misión durante todo el escenario de la misma. El proceso seguido se detalla en la sección 8.2.

Figura 8.8. Coste del control difuso escalado usando el coste del PID.

8.2. Control Difuso-Adaptativo

Como ya se ha visto en el Test 3 (sección 8.1.3), el control difuso no cumple con el criterio de precisión impuesto en la calibración (ec. (7.9)) a

bajas alturas. Esto se debe principalmente a que existe un par de perturbación que tiene una forma exponencial con la altura (par producido por la resistencia aerodinámica como se ve en la sección 2.2.1). Además, el proceso de calibración se ha llevado a cabo en un entorno de simulación en el que dicha perturbación no es significativa. Si se repite el proceso de calibración en una órbita a doscientos kilómetros se obtiene una calibración que cumple el criterio de precisión para toda la misión pero a costa de incrementar la función de coste (i. e. aumentando el consumo energético global). Dado que éste es un factor significativo en el control y que minimizarlo al máximo es un objetivo de esta tesis, se diseña un control Difuso-Adaptativo (en adelante DA) que obtiene lo mejor de ambas calibraciones.

El algoritmo DA tiene en cuenta las perturbaciones estimadas por el DKE y modifica las funciones de pertenencia del control difuso original de manera acorde (figura 8.9).

Figura 8.9. Esquema del control DA. T_{dj} es la perturbación estimada en el eje j .

En este caso, las perturbaciones estimadas por el DKE son normalizadas usando un rango de valores típico en la misión considerada. Después el valor normalizado del par de perturbación se cataloga como "Grande" o "Pequeño" (figura 8.10 izquierda) y modifica al control original haciéndolo más "parecido a órbita baja" o más "parecido a órbita alta" (figura 8.10 derecha). El control original ve modificadas sus funciones de pertenencia, en concreto se modifican los valores discretos que hacen de funciones de pertenencia de la variable de salida, alterando así su comportamiento y logrando que éste sea un híbrido entre ambas calibraciones. Los valores de salida que aparecen en

la figura 8.10 derecha han sido ajustados para que la salida máxima y mínima se corresponda con la necesaria según la calibración en órbita de baja altura o en órbita alta respectivamente. Esta versión modificada es la llevada implementada a bordo el satélite QBITO.

Figura 8.10. Funciones de pertenencia del control DA: a la izquierda las funciones de pertenencia en la variable de entrada y a la derecha para la variable de salida.

Solo los ejes Y_B y Z_B se actualizan para usar el algoritmo DA. El eje X_B no tiene perturbaciones significativas dependientes de la actitud por lo que la salida del algoritmo DA es la misma que la salida del algoritmo difuso original. Para estudiar el desempeño de este nuevo algoritmo, se repite en la sección 8.3.1 el Test 3 usando ahora el algoritmo DA.

8.3. Calibración a baja altura

Antes de repetir el Test 3 se considera interesante calibrar de nuevo el PID a baja altura (igual que se ha hecho previamente con el difuso para el diseño del algoritmo DA) para comparar el algoritmo original con esa nueva versión. Los resultados de esta calibración, así como de la calibración del control difuso a baja altura, se recogen en la tabla 8.1. Es interesante mencionar que al calibrar el PID a baja altura las tres ganancias integrales salen tan pequeñas que se dejan en cero.

		K_p	K_d	K_i
PID	X	5E-5	-1.4E-3	0
	Y	2.5E-5	-1.1E-3	0
	Z	2.5E-5	-1.1E-3	0

		K	$K_{\Omega OB}$	K_{T_c}	K_{ω}
difuso	γ	34.38	26.18	4.2E-6	-
	σ	5.73	12.74	9E-6	-0.48
	μ	52.14	43.63	1.9E-5	-

Tabla 8.1. Ganancias del PID y del control difuso calibrados a baja altura.

Se observa (ver figura 8.11) que la nueva calibración a baja altura cambia la forma de las curvas, haciéndolas menos sensibles a la altura. Sin embargo, el control difuso sigue siendo más eficiente (entre un 15% y un 55% dependiendo del eje considerado).

Figura 8.11. Ratio de costes para maniobras de cincuenta grados en cada uno de los ejes, entre el control difuso original y dos controles de tipo PID, uno optimizado para baja altura y otro para alta (calibración original).

8.3.1. Test 3.2

Repitiendo el Test 3 usando el algoritmo DA y comparando con el PID calibrado para una órbita alta (para poder comparar con las gráficas originales del Test 3) se obtienen los siguientes resultados en error (figura 8.12 y figura 8.13) y en ratio de costes (figura 8.14 y figura 8.15). Como referencia se han pintado también las curvas correspondientes al comportamiento del control difuso original, tanto con las ganancias de la calibración original a gran altura como la calibración a baja altura hecha en la sección 8.2. Se observa como el algoritmo DA mejora a ambas calibraciones en términos de coste mientras hace una transición suave entre las dos curvas de error residual.

Figura 8.12. Error residual del control DA en cabeceo comparado con el original, calibrado para órbitas altas y para órbitas bajas.

Figura 8.13. Error residual del control DA en guiñada comparado con el original, calibrado para órbitas altas y para órbitas bajas.

Figura 8.14. Ratio de costes del control DA en cabeceo comparado con el original, calibrado para órbitas altas y para órbitas bajas.

Figura 8.15. Ratio de costes del control DA en guiñada comparado con el original, calibrado para órbitas altas y para órbitas bajas.

Puede resultar chocante que el ratio de costes sea mejor que cualquiera de las dos calibraciones en las que se basa el DA mientras que el error residual es virtualmente idéntico al obtenido por cada una de ellas en el límite correspondiente (la calibrada a baja altura en órbitas bajas y lo mismo para la correspondiente a órbitas altas). Esto se explica porque a bajas alturas, solo se comporta "parecido a órbita baja" cuando el par es grande, por lo que para actitudes cercanas a la posición nominal, el control DA se comporta como la calibración para órbita alta que es más eficiente. Por lo tanto, para pequeñas correcciones no necesita actuar con la misma intensidad con que lo haría para correcciones grandes.

8.3.2. Test 4

En las secciones anteriores de este capítulo se ha comprobado como el control de tipo DA proporciona un control eficiente y preciso, cumpliendo con los requisitos en todo el rango de alturas de la misión y a un coste significativamente menor que el correspondiente a un control PID de referencia. Sin embargo, hasta ahora solo se ha estudiado la respuesta de

ambos controles sometidos a comandos estáticos (Test 1 y Test 2) o a señales de tipo escalón (Test 3). Se necesita un testeo más amplio que garantice que el control va a desempeñarse bien en un amplio rango de situaciones que pueden surgir a lo largo de la misión. Para intentar generalizar estos escenarios, se han ideado dos test en los que se estudia la respuesta a condiciones de contorno dinámicas: primero el Test 4 estudia cómo cada uno de los algoritmos de control es capaz de contrarrestar unas condiciones iniciales dinámicas, por otro lado, el Test 5 (ver Sección 8.3.3) analiza el comportamiento al intentar seguir actitudes de referencia variables en el tiempo.

Hasta ahora todas las simulaciones se han hecho partiendo de condiciones nominales estables. Sin embargo, en el caso más general posible, el satélite parte de unas condiciones iniciales arbitrarias: rotando de manera desconocida tras ser desplegado desde el lanzador. Por lo tanto, ser capaces de controlar dicho estado dinámico es fundamental. Es importante matizar que en el SW de control de actitud embarcado en QBITO existe un módulo de control específico encargado de frenar al satélite después del despliegue y antes de pasar el relevo al algoritmo DA principal. Este módulo de control se basa en un algoritmo con nombre propio llamado B-dot (Lovera, 2015), que no se va a estudiar aquí. Sin embargo, aunque que el grueso de la estabilización corra a cargo de dicho algoritmo, éste nunca acaba con el satélite perfectamente estacionario. Por el contrario, acaba con el satélite girando lo suficientemente lento como para el que control principal sea capaz de controlar al satélite. Por lo tanto, las condiciones iniciales del control embarcado serán una actitud arbitraria y una rotación a velocidad constante respecto a un eje también arbitrario. Este hecho obliga a que se deba verificar que el control es capaz de estabilizar al satélite partiendo de condiciones iniciales dinámicas no nulas.

Aunque es sabido que en un sólido rígido girando alrededor de una dirección aleatoria se producen efectos de transferencia de momento entre ejes, se considera interesante estudiar el tiempo requerido por cada uno de los algoritmos de control para frenar al satélite partiendo del CubeSat girando alrededor de tan solo uno de sus ejes. Después se repite el proceso simulando el comportamiento del satélite con condiciones iniciales dinámicas en el resto de ejes por separado. En esta simulación se han probado rotaciones iniciales de hasta cuarenta grados por segundo a pesar de que en la misión real no se esperan nunca velocidades mayores a diez grados por segundo. Esto se hace para tener suficiente confianza en el desempeño del controlador que está embarcado en QBITO. Los resultados de este estudio se muestran en la figura

8.16 donde se observa que existen dos tendencias diferentes. La primera, en la región baja de la gráfica sin apenas dependencia con la velocidad inicial, donde están el control de balanceo mediante PID y los de cabeceo y guiñada mediante DA. La segunda región, donde están el resto de casos, tiene una dependencia aproximadamente lineal con la velocidad de rotación inicial.

Figura 8.16. Tiempo requerido por ambos algoritmos de control para alcanzar una actitud nominal estable tras empezar con condiciones iniciales dinámicas en cada uno de los ejes principales del satélite.

Es interesante explicar la razón por la que en la figura 8.16 se estudia el tiempo hasta la convergencia y no el error aparejado a la solución final. Cuando se parte de unas condiciones iniciales dinámicas, lo primero que hace el control es intentar frenar del todo al satélite. Una vez que el satélite está casi frenado, se solapan el proceso de frenar al satélite y el de alcanzar la actitud nominal. Esta última etapa es la que se ha estudiado en secciones anteriores y cuya convergencia ya se conoce. Por lo tanto lo que ahora se quiere estudiar es el proceso inicial por el cual el control frena al satélite. En este test no se ha analizado el coste aparejado a cada maniobra porque, al tener tiempos característicos muy dispares, los costes asociados no son comparables.

Es importante recalcar que en menos de una órbita (de hecho en menos de 30 minutos o un tercio del periodo orbital aproximadamente) el satélite se ha estabilizado sin importar las condiciones iniciales impuestas. Además, vistas las tendencias es de esperar que rotaciones más rápidas incrementen proporcionalmente el tiempo necesario para estabilizarse, pero que el satélite siga siendo controlable de todos modos.

8.3.3. Test 5

En el último test de esta verificación se estudia la respuesta del satélite cuando se impone como objetivo, a cada uno de los ejes por separado, una señal sinusoidal de amplitud veinte grados. La frecuencia de dicha señal se incrementa progresivamente de repetición en repetición para estudiar la respuesta en frecuencia del control. Para evitar problemas de acoplamiento en frecuencia, así como otros efectos inesperados se decide seguir con la filosofía seguida hasta ahora y hacer una simulación para cada frecuencia, sin variar la frecuencia durante la simulación. Aunque en la misión QBITO no se prevén actuaciones fuera de la actitud nominal, igual que se ha hecho en las secciones previas, se quiere caracterizar el control difuso con un poco más de detalle. Una maniobra típica en un satélite de observación terrestre consiste en apuntar hacia unas coordenadas concretas sobre la superficie terrestre. En estos casos la actitud comandada es variable en el tiempo y se puede aproximar por una señal sinusoidal de un periodo desconocido (relacionado con la altura de la órbita y otros parámetros de la misión en cuestión).

En la figura 8.17 se ha representado el error como porcentaje de la señal original una vez alcanzado el estacionario del sistema. Hay que matizar que no se ha distinguido entre errores en amplitud y errores en fase. Esto quiere decir que dos señales de idéntica amplitud y frecuencia, pero cuyas fases no sean idénticas presentan un error significativo, proporcional (pero no lineal) al desfase. Como ejemplo, si el satélite sigue exactamente la señal de excitación desfasado $1/12$ rad, el error que se representaría es de cerca del 16%. Como además es imposible que el satélite siga exactamente la señal de excitación (el control solo actúa cuando existe un error), en el mejor de los casos, el control conseguirá una respuesta igual que la señal de excitación, pero con un cierto desfase, mayor cuanto más rápida sea la señal de excitación. Por lo tanto, errores de hasta un 20% se pueden considerar como razonables en este test. Una vez matizado esto, si se analiza la gráfica se

observa que el PID se comporta mejor en términos generales que el DA, especialmente para el rango de alta frecuencia, donde el DA resulta inefectivo en un rango mayor de frecuencias. Este comportamiento entra dentro de lo esperable ya que en el método de calibración escogido se primaba el coste por encima de la rapidez de respuesta. De hecho, como se observa en la figura 8.3 el control difuso es, en términos generales, más lento que el PID. En concreto, los ángulos de balanceo y guiñada controlados por el difuso tardan significativamente más que sus homólogos controlados mediante PID en llegar a un entorno cercano a la solución (tardan alrededor de cincuenta segundos más en llegar al entorno de dos grados y medio alrededor de la solución). Esto es así porque este tipo de comandos no se esperan en la misión real y por lo tanto el proceso de calibración no ha intentado optimizar esta faceta del comportamiento de los controladores. Sin embargo, se observa como para frecuencias correspondientes a periodos superiores a los 200 segundos el error es menor del 20%, lo que equivaldría a un desfase de menos de $1/10$ rad, siendo considerado válido para la misión. Parece interesante poner en contexto este valor siguiendo con el ejemplo anterior de un satélite que desea observar un punto determinado sobre la superficie terrestre. En cada sobrevuelo el satélite debe rotar 180° en un tiempo similar al del pase sobre ese punto. Esto corresponde a una maniobra sinusoidal con periodo del orden de 1.000 s.

Figura 8.17. Respuesta en frecuencia para una excitación sinusoidal de veinte grados de amplitud.

8.3.4. Conclusiones

Los test efectuados hasta este punto muestran que el control difuso es significativamente más económico en términos de función de coste que el control PID de control. Sin embargo, la calibración efectuada ha optimizado tanto en coste que el error se dispara fuera de los límites impuestos por la misión en cuanto los pares de perturbación aumentan. Se ha estudiado también el uso de una calibración más restrictiva que obligue al cumplimiento de los requisitos durante la fase más conservadora de la misión. Esta calibración es menos económica que la inicial dado que ahora en órbitas altas, donde las perturbaciones son menores, el control se muestra demasiado agresivo. Por último se ha decidido implementar un algoritmo adaptativo, llamado Difuso-Adaptativo (DA), para realizar una transición suave entre una calibración u otra dependiendo del entorno así como de la actitud instantánea. Los test efectuados con ese nuevo algoritmo indican que el algoritmo DA es tan preciso o más que el PID de referencia, siendo significativamente más eficiente. En concreto el algoritmo DA es capaz de mantener el error residual por debajo de 0.01 rad consumiendo hasta un 70% menos de energía en maniobras sencillas.

9. ANÁLISIS MULTIVARIABLE TIPO MONTE CARLO

En los capítulos 7 y 8 se ha considerado que se conocían todos los parámetros que aparecen en el DKE. Sin embargo hay parámetros que se han estimado, unos por ser imposible la medición directa en etapas tempranas de diseño (v. g. la matriz de inercia del satélite completo) y otros que tienen tolerancias debido a los procesos de fabricación y montaje. Todos estos parámetros tienen efecto en las simulaciones y por extensión en el control desarrollado. Además, como también se ha indicado, los actuadores han sido modificados, tanto a nivel *hardware* como del *firmware* incorporado. Por último, en las simulaciones efectuadas en el capítulo 8 se ha considerado que el punto espacial en el que se inicia la simulación es irrelevante, dado que los actuadores han sido idealizados y no dependen del campo magnético. En cualquier caso hay que estudiar cómo se comportan ambos controles en un amplio rango de configuraciones y escenarios, para tener confianza en el buen desempeño del SW embarcado en QBITO. Además, esto permite extrapolar los resultados previos (sección 8.1) a distintas configuraciones de satélites, lo que facilita el uso de estos algoritmos en un amplio espectro de misiones (JCGM, 2008).

El objetivo buscado cumple perfectamente con el ámbito de aplicación del método Monte Carlo (Metropolis, 1949). Dicho método estudia estadísticamente los problemas mediante la variación aleatoria de ciertos parámetros (Armendáriz, 2015). Cuantas más veces se repita el test, más fiables serán las conclusiones que se obtengan, pero a su vez, más recursos se necesitan, tanto para efectuar las simulaciones como para analizar los resultados obtenidos. El método Monte Carlo se puede esquematizar como se muestra en la figura 9.1.

Figura 9.1. Esquema del método de Monte Carlo. t es el tiempo simulado y N es el número de repeticiones y puede ser tan grande como se necesite.

En esta tesis, el número de iteraciones hechas está en el entorno de $N=38.000$ repeticiones, siendo cada iteración dos simulaciones (DA y PID) de 1.000 segundos. La duración se ha fijado, conservativamente, en 1.000 segundos para que ambos controles tengan tiempo suficiente de converger en

todas las configuraciones y escenarios estudiados. El principal factor limitante a la hora de fijar ese número ha sido el tiempo necesario para las simulaciones. Como referencia, cada simulación requiere aproximadamente entre 20 y 30 segundos en un ordenador de oficina normal.

9.1.1. Escenario

Como ya se ha mencionado, la base del método Monte Carlo consiste en variar de manera aleatoria una serie de parámetros. En este estudio, los parámetros que van a ser variados aleatoriamente son los siguientes:

- La matriz de inercia del satélite.
- El momento de inercia de la rueda de momento sobre su eje.
- El dipolo residual del satélite.
- La posición orbital tridimensional.
- Las condiciones iniciales, tanto dinámicas como estáticas del satélite.

La justificación para seleccionar estos parámetros son las posibles tolerancias de fabricación y montaje (elementos 1-3) así como posibles escenarios previsibles en la misión (elementos 4 y 5). A continuación se describe el modelo empleado para cada uno de los parámetros.

(i) La matriz de inercia del satélite.

La matriz de inercia se ha variado tomando como referencia la matriz originalmente estimada de QBITO. Los nuevos valores para los momentos de inercia se han obtenido usando distribuciones aleatorias normales centradas en los valores originales (ver figura 9.2). La varianza de las distribuciones normales se ha hecho lo más grande posible considerando únicamente los momentos de inercia positivos y el valor I_{zz} principalmente por encima de I_{xx} para que se mantenga la estructura global de la matriz de inercia en términos generales:

$$\begin{aligned} I_{xx} &= N(4, 1.1) \\ I_{yy} &= N(12, 3.5) \\ I_{zz} &= N(14, 3.0) \end{aligned} \tag{9.0}$$

Figura 9.2. Distribución de probabilidad de los nuevos momentos de inercia.

Además, se ha tenido la precaución de que los momentos principales de inercia sean siempre mayores que los productos cruzados dado que en el satélite los momentos de inercia van a ser siempre mucho mayores que los productos cruzados. Para ellos se ha seguido la siguiente norma:

$$I_{ii} = \begin{cases} I_{ii} + I_{lat} & \text{si } I_{ii} \leq I_{lat} \\ I_{ii} & \text{si } I_{ii} > I_{lat} \end{cases} \quad (9.1)$$

Donde I_{lat} es el mayor de los productos de inercia cruzados:

$$I_{lat} = \max_{i \neq j} I_{ij} \quad (9.2)$$

(ii) El momento de inercia de la rueda de momento sobre su eje.

Se usa la misma filosofía seguida para variar el momento de inercia de la rueda respecto a su eje de giro. En este caso se han evitado los valores demasiado pequeños ya que comprometen la controlabilidad del sistema:

$$I_w = N(2, 0.3) \quad (9.3)$$

(iii) El dipolo residual del satélite.

El dipolo residual no sigue la misma pauta porque no existe un valor teórico. Sin embargo, se ha considerado interesante centrarlo en un valor no nulo. Además, dada su naturaleza infinitesimal, se ha preferido usar la distribución normal como exponente para que la distribución obtenida abarque varios ordenes de magnitud y sea siempre positiva.

$$D_i = 10^{N(-7, 1)} \quad (9.4)$$

Como ya se ha indicado en la sección 6.2.1, D denota la normal del dipolo magnético del satélite, siendo D_i el dipolo residual del satélite en el eje i .

(iv) La posición orbital tridimensional.

Como ya se ha mencionado, se considera interesante variar la posición tridimensional de inicio de las simulaciones dado que el comportamiento del control puede verse afectado. Esto es debido a que las condiciones ambientales no son homogéneas alrededor de la Tierra (v. g. el ángulo relativo del vector campo magnético varía desde casi paralelo al eje $\mathbf{X}_0$ hasta casi ortogonal). Para cubrir la máxima extensión posible de órbita y a su vez seguir siendo aleatorio, se han variado los parámetros orbitales de Ascensión Recta del Nodo Ascendente y la Anomalía Verdadera usando una función aleatoria pura (distribución de probabilidad plana), todo ello sin modificar el resto de parámetros orbitales. Estos dos parámetros se pueden convertir en una distribución aleatoria en latitud y longitud como se observa en la figura 9.3.

Figura 9.3. Distribución espacial de los puntos iniciales para las simulaciones.

(v) Las condiciones iniciales, tanto dinámicas como estáticas del satélite.

La actitud inicial del satélite es un giro arbitrario entre 0 y 2π en cada uno de los ejes del satélite. Así mismo, la velocidad de rotación en cada uno de esos ejes es una variable aleatoria entre -1.5 y 1.5 grados por segundo. Estos valores se han tomado porque representan un amplio espectro de situaciones, pero a su vez, pueden ser corregidos en menos de 1000 segundos, que es el tiempo de simulación usado. Además, al imponer condiciones iniciales no nulas y exigir que se alcance la actitud nominal estamos minimizando el impacto de los efectos de acoplamiento entre ejes (ver figura 8.7).

Con cada juego de parámetros aleatorios, se efectúa una simulación de 1.000 segundos en la que se impone como objetivo la actitud nominal (i. e. los ejes del sistema BRF coinciden con los del sistema ORF). De cada simulación se guardan en memoria una serie de parámetros de salida que son los que nos van a servir para analizar estadísticamente ambos algoritmos de control.

- Error instantáneo tridimensional cada diez segundos.
- Error residual.
- Coste total de la estabilización.

Es importante recalcar que todos los errores y costes que se van a manejar han sido calculados usando las expresiones de las ec. (8.1) y ec.(8.2)

La primera de las variables de salida (el error instantáneo) contiene en sí misma información relativa a toda la simulación realizada. Sin embargo, dado el volumen de simulaciones, el uso de esta variable para realizar análisis globales se estima excesivo. Por ello, esta variable se utiliza para analizar como es, *grosso modo*, el comportamiento del control para esa configuración así como ver el comportamiento temporal de conjuntos puntuales de parámetros (v. g. como se comporta el satélite cuando las variables de entradas tienen módulo mínimo). Sin embargo, al ser una serie temporal muestreada a 0.1 Hz, no se usa para analizar numéricamente los resultados porque, por un lado tiene demasiada información (cien valores por simulación) pero por el opuesto, puede haber perdido información relevante (v. g. oscilaciones a 0.1 Hz en el final de la simulación aparecerían como líneas horizontales).

Para análisis numérico se usa una medida del error promediado calculado como:

$$e^* = \max \left(\left| \sum_{i=950}^{1000} \frac{\tilde{e}(i)}{50} \right|, \left| \sum_{i=900}^{1000} \frac{\tilde{e}(i)}{100} \right|, \left| \sum_{i=850}^{1000} \frac{\tilde{e}(i)}{150} \right| \right) \quad (9.5)$$

Como se observa, de cada simulación se obtienen tres medidas independientes del error, quedándonos con la mayor de ellas. Además de estas tres medidas, también se obtiene la desviación típica en los últimos cincuenta segundos. Estas dos variables (error residual máximo y desviación típica) se toman para intentar minimizar el riesgo de que respuestas oscilantes alrededor del cero, que tendrían valores ridículamente buenos, sean incorrectamente valoradas.

Por último, la integral de la función de coste, definida en la ec. (7.11), se usa como una medida de la eficiencia del control en ese escenario particular.

$$C = \int_0^{1.000} C(t) dt \quad (9.5)$$

A nivel operativo, y para reducir el impacto de interrupciones en la ejecución de las múltiples simulaciones, se decide grabar en disco cada diez simulaciones. Además cada fichero de datos almacena información de solo mil simulaciones, generándose múltiples ficheros a lo largo de todo el test. Después todos los resultados son cargados en memoria de manera secuencial para su análisis estadístico.

9.1.2. Análisis de datos

El primer paso que se ha dado consiste en analizar si los parámetros de entrada que se han escogido son independientes y ver si los parámetros de salida tienen alguna correlación evidente con los de entrada. Para ello se ha creado la figura 9.4 donde se han representado en la diagonal los histogramas de cada una de las variables, tanto de entrada como de salida. En las variables de salida se han superpuesto los histogramas del PID y del difuso. Fuera de la diagonal se encuentran las gráficas de dispersión que estudian la correlación entre las variables correspondientes a esa fila y columna. En cian están pintados los puntos de correlación entre variables de entrada. Los puntos en azul oscuro corresponden a la correlación entre variables de entrada y las salidas para el PID, siendo los puntos verdes los homónimos para el control difuso.

Debido a la escasez de espacio útil para las etiquetas de las variables, los nombres utilizados han tenido que ser simplificados al máximo. A

continuación se van a detallar los parámetros representados de izquierda a derecha. Los seis primeros parámetros son todos los correspondientes a la matriz de inercia del satélite. A continuación está el momento de inercia de la rueda de momento respecto a su eje de giro. El siguiente elemento (α_0), el séptimo por la izquierda, es el error tridimensional definido por la ec. (8.1) cuando se calcula el cuaternión del satélite con las variables Φ , θ , ψ . Éstas variables son los giros arbitrarios alrededor de cada uno de los ejes del satélite que se han impuesto como parámetros del test. A continuación se encuentran las componentes de la velocidad de rotación respecto a ORF así como su módulo. Los últimos parámetros son las componentes del dipolo residual, la Ascensión Recta del Nodo Ascendente y la Anomalía Verdadera. Por último, a la derecha de todo están los logaritmos decimales del error y del coste, definidos como en la ec. (9.5) y la ec. (9.5) respectivamente.

Figura 9.4. Matriz de correlaciones entre variables de entrada y entre variables de entrada y variables de salida de las simulaciones de Monte Carlo.

A simple vista no se observa ningún tipo de correlación entre las variables de entrada. Sin embargo, sí que se observan algunos comportamientos peculiares en la zona de gráficos color cian. En concreto se comprueba que los productos de inercia I_{xy} e I_{yz} muestran una correlación casi perfecta. Esto se debe a que por un error en la generación aleatoria de estas variables, la variable I_{xy} se ha reajustado incorrectamente sumándole I_{yz} , y dado que I_{yz} es mucho mayor que I_{xy} ésta última es siempre prácticamente igual que I_{yz} . Se observa que la variable α_o tiene una cierta correlación con los ángulos de Euler iniciales. Esto se debe a que α_o es el error tridimensional definido por la ec. (8.1) cuando se calcula el cuaternión con las variables Φ , θ , ψ .

A continuación, se ha analizado la respuesta general obtenida por ambos controles en todas las simulaciones. En la figura 9.5 se han representado superpuestas las respuestas obtenidas en las 38.000 simulaciones efectuadas.

Figura 9.5. Error tridimensional instantáneo. Cada línea representa una simulación independiente con parámetros aleatorios diferentes.

Dada la gran aglomeración de líneas, sobre todo en la parte izquierda de las gráficas de la figura 9.5, es complicado sacar conclusiones relevantes. Sin embargo, este hecho es en sí mismo un resultado interesante: la gran mayoría

de simulaciones convergen aparentemente bien en menos de seiscientos segundos. Sin embargo, existen configuraciones en los que la actitud conseguida no es la nominal. Estos casos se estudiarán en detalle más adelante.

Además, parece que el control PID es más rápido que el difuso, lo que sería coherente con los resultados obtenidos previamente (sección 8.1). Para analizar el tiempo que necesita cada control en llegar "cerca" de la solución se han utilizado las siguientes variables:

$$\begin{aligned} e^* &< 10^{-3} \\ \Delta e^* &< 10^{-3} \end{aligned} \tag{9.6}$$

Siendo Δe^* la diferencia entre el valor del error en un instante y en el instante inmediatamente anterior. En la figura 9.6 se observa como en más de la mitad de los casos (54%), el control difuso es más rápido que el PID.

Figura 9.6. Tiempo necesario hasta estar "cerca" de la solución (ec. (9.6)).

Un hecho destacable de la figura 9.5 es que existen configuraciones fallidas, entendidas como aquellos parámetros que causan que el control no converja a la solución nominal. Antes de analizar dichos casos, se presentará un gráfico más (figura 9.7), que sirve para analizar este comportamiento anómalo.

Se observa que la gran mayoría de casos tienen errores inferiores a 0.01 radianes usando cualquiera de los dos algoritmos de control. Además, también se observa como el control difuso está ligeramente desplazado a la izquierda, lo que sugiere que el control difuso puede ser ligeramente más preciso que el control PID en términos generales. Sin embargo, como se

esperaba, hay un número significativo de casos en los que el error residual es demasiado grande, dándose igualmente más incidencia de estos casos fallidos cuando se usa control PID que cuando se usa el difuso.

Figura 9.7. Histograma del orden de magnitud del error residual.

Analizando esos casos en los que el control no es capaz de converger de manera adecuada, se llega a la conclusión que la causa principal es que alguno de los tres momentos principales de inercia es muy pequeño. Estos momentos de inercia pequeños representan casos de sólidos degenerados (en el límite representan una distribución másica unidimensional en un sólido tridimensional). Por lo tanto la primera conclusión a la que se llega es que ambos controles pierden la capacidad de controlar al satélite cuando alguno de sus momentos de inercia es demasiado pequeño.

Estudiando los conjuntos de parámetros descartados (figura 9.8) se observa cómo no todos los momentos de inercia degenerados afectan por igual a ambos algoritmos de control. Los puntos a la izquierda de las líneas verticales son configuraciones descartadas por ser degeneradas en alguno de los otros dos momentos de inercia. De hecho, el momento I_{XX} afecta principalmente al control PID, el I_{ZZ} al control difuso y el I_{YY} afecta a ambos de forma similar. Esto se observa en la figura 9.8 ya que, en la gráfica

superior izquierda, los puntos a la izquierda del umbral, representado por la línea azul vertical, se agrupan en dos zonas, la zona superior de errores significativos en el control PID y la inferior con errores aceptables en el control difuso. En la gráfica inferior izquierda se observa el efecto opuesto, en la parte superior de los puntos descartados se agrupan los valores correspondientes al control difuso y en la inferior los del control PID. En la gráfica superior derecha se observa que los puntos correspondientes a ambos controles están mezclados por lo que no hay una influencia clara.

Además hay que mencionar que, coherentemente con lo visto en la figura 9.7 los errores de los puntos descartados tienen errores del orden de -2 en general para el control difuso y hasta orden 0 para el PID.

Figura 9.8. Momentos de inercia descartados. Las líneas azules verticales representan el umbral seleccionado para cada momento de inercia.

Una vez eliminados los datos correspondientes a las simulaciones de estos momentos degenerados ambos algoritmos de control se comportan correctamente (figura 9.9): todas las simulaciones convergen a la actitud nominal.

Figura 9.9. Error tridimensional instantáneo de todas las simulaciones salvo las correspondientes a los momentos de inercia degenerados.

Figura 9.10. Histograma del orden de magnitud del error residual para las simulaciones no degeneradas.

El comportamiento general de ambos controles no se ha visto afectado por haber quitado estos momentos de inercia extremos como se aprecia en la figura 9.9. Este efecto puede estudiarse con más detalle repitiendo el cálculo del histograma para las simulaciones restantes (figura 9.10). En éste se observa como la forma general de ambas gráficas no ha cambiado apenas, salvo en el extremo derecho donde han desaparecido los errores significativos que aparecían antes. Sigue apreciándose que, en general, el control difuso parece más preciso que el control PID.

Hasta ahora solo se ha analizado el comportamiento de ambos controles por separado, sin saber si un mal control PID va unido a un mal control del difuso o por el contrario hay escenarios en los que es mejor uno que el otro. Para analizar más a fondo este efecto vamos a estudiar, para cada simulación, el ratio de costes y el ratio de errores. Dado que los órdenes de magnitud son muy dispares se considera interesante analizar el logaritmo decimal de ambos cocientes como se observa en la figura 9.11.

Figura 9.11. Dispersión de los logaritmos de los cocientes de errores y costes entre ambos controles. Las líneas punteadas marcan los valores medios.

Aplicando un test de bondad del ajuste se demuestra cómo ambas nubes de punto se asemejan a las siguientes distribuciones normales:

$$\log_{10}\left(\frac{C_{DA}}{C_{PID}}\right) = \log_{10}(\tilde{C}) \sim N(-0.171, 0.294) \quad (9.7)$$

$$\log_{10}\left(\frac{e_{DA}^*}{e_{PID}^*}\right) = \log_{10}(\tilde{e}) \sim N(-0.393, 0.283) \quad (9.8)$$

De ambas ecuaciones (ec. (9.7) y ec. (9.8)) se deducen las siguientes conclusiones:

- Existe más de un 90% de probabilidad para que el control difuso sea más preciso que el PID en el mismo escenario.
- Existe más de un 70% de probabilidad para que el control difuso sea más eficiente que el PID en el mismo escenario.
- El control PID es en general un 50% más caro y hasta un 250% menos preciso que el control difuso en un amplio conjunto de configuraciones.

Hasta ahora hemos analizado el coste y el error por separado, sin analizar si un descenso en el error relativo lleva aparejado un incremento en coste o al revés. Por lo tanto el siguiente paso es correlacionar el ratio de errores y el ratio de costes (figura 9.12). Como antes, se va a seguir utilizando el logaritmo de ambos ratios pues permite un análisis más global de los datos.

En la figura 9.12 se han dibujado también las líneas de nivel que indican el porcentaje de puntos que se haya dentro de dichas líneas. Esto se considera interesante porque dentro de la zona central los puntos se superponen y es complicado hacerse a la idea de la distribución de los mismos. El centro de dichas curvas cae en el punto $[-0.39, -0.17]$ que son los valores medios obtenidos con anterioridad (ec. (9.7) y ec. (9.8)).

Figura 9.12. Dispersión de los logaritmos de los ratios de errores y costes con las líneas de nivel de densidad superpuestas. Entre dos curvas consecutivas se encierran aproximadamente el 10% de los puntos.

Como referencia, se han calculado los porcentajes de puntos en cada uno de los cuadrantes. El primer cuadrante representa los controles que son a la vez más precisos y más económicos con el PID que con el difuso. El segundo cuadrante tiene los puntos en los que el control difuso es más preciso pero menos eficiente. En el cuarto cuadrante se tienen los opuestos, aquellos en los que el control difuso es menos preciso pero más eficiente. Por último, en el tercer cuadrante se tienen los puntos en los que el control difuso es más preciso y eficiente que el PID. Tan solo un 2.9% de las simulaciones muestran que el PID es mejor en general que el difuso, mientras que en un 65.2% de ellas el difuso es mejor que el PID en todos los sentidos estudiados.

9.1.3. Conclusiones

En este capítulo se ha empleado un análisis estadístico aleatorio para comparar el comportamiento de los controladores PID y difuso frente a una serie de incertidumbres en el diseño y el proceso de fabricación y montaje así como posibles escenarios de la misión. Utilizando el método de Monte Carlo,

efectuando más de 38.000 simulaciones se han obtenido una serie de datos globales que caracterizan y comparan mejor ambos controles. Este estudio ha demostrado que el control difuso es mejor que el PID en un amplio rango de circunstancias, siendo en media el PID hasta un 50% menos eficiente y un 250% menos preciso. Además, analizando los resultados de ambos algoritmos de control para cada una de las simulaciones, se observa que en un 65.2% de los escenarios posibles el algoritmo difuso es más preciso y más eficiente que el algoritmo de control PID. La situación opuesta se da sólo en un 2.9% de los escenarios estudiados. Por último, en el 31.9% de los escenarios analizados el control difuso mejoraba al control PID en solo una de las variables estudiadas, siendo peor en la otra.

10. SOFTWARE EMBARCADO

Hasta ahora, todo el control del ADCS se halla en forma de código integrado dentro del simulador que corre en el entorno Matlab-Simulink. Sin embargo, el SW embarcado debe ejecutarse embebido en un microprocesador, capaz de interactuar con todos los periféricos del subsistema así como con el resto de subsistemas. La primera diferencia evidente es que el lenguaje de programación utilizado para el SW embarcado debe ser C en lugar de Matlab. Además, es necesario cambiar la estructura del control para compartimentarlo así como añadir módulos no implementados en el simulador (i. e. módulos de determinación y de interfaz con el HW). Por último, y dadas las características del SW embarcado, se decide utilizar un propagador orbital diferente, más adecuado a las operaciones en órbita.

10.1. Diseño conceptual

El primer paso a realizar antes de exportar el código existente en Matlab, es hacer un diseño esquemático del SW embebido que se quiere conseguir. Existen muchas maneras de describir SW pero se ha optado por seguir el estándar *Unified Modeling Language* (UML por sus siglas en inglés). De entre todos los esquemas del estándar, y dado que el código a desarrollar va a ser monohilo y no orientado a objetos, se ha decidido incluir sólo los modelos de secuencia y el de nodos de la actividad. Antes de profundizar más, se considera interesante aclarar alguno de los términos que se acaban de utilizar. A pesar de que tanto el lenguaje como la arquitectura del procesador admiten la ejecución de SW multihilo (v. g. el OBC utiliza más de veinte hilos), se ha optado por implementar el código del ADCS en modo monohilo. En este entorno se entiende hilo como una secuencia de tareas encadenadas que pueden ser ejecutadas paralelamente en modo multitarea. Esto quiere decir que el SW del ADCS se ejecuta de manera determinista siguiendo una secuencia predefinida e inmutable con la única salvedad de la gestión de comandos y telemetrías, que se ejecutan como interrupción por necesidades del protocolo de comunicaciones I2C.

Como ya se ha mencionado, se va a describir el SW embarcado mediante una serie de esquemas del estándar UML. Primero se va a hacer uso del modelo de secuencia de dicho estándar. Este modelo se lee de arriba abajo y representa las interacciones entre los distintos actores que se consideren. Estos actores pueden ser físicos (v. g. el ADCS, el OBC, operador en tierra

etc.) o pueden ser virtuales (i. e. interacción entre distintas funciones de un mismo código). La secuencia que se describe es inmutable dado que, como se ha mencionado, el SW del ADCS es monohilo.

En general, se va a partir de lo más general y después se irá viendo en detalle cada uno de las partes más importantes. En la figura 10.1 se han representado las dos primeras ejecuciones del ADCS tras el despliegue desde el lanzador.

Tras el despliegue, por motivos de seguridad para con el módulo de despliegue (sección 11.2.2), hay una pausa de treinta minutos antes del inicio de la ejecución principal del SW, así como del despliegue de antenas. Después, QBITO emite un mensaje corto (*beacon*) cada treinta segundos y se queda en *standby* a la espera de instrucciones desde tierra. En lo que respecta al ADCS, éste permanece apagado hasta que se enciende desde tierra. La primera instrucción que aparece es la de " Empezar ADCS SW " que lleva aparejadas una serie de acciones por parte del OBC y el EPS, que culminan con el ADCS activo y ejecutando el SW indicado. Hay que resaltar que aunque solo se ha etiquetado como "*Bootloader*" la parte inferior, el mismo bloque se encontraría en la parte superior, siendo lanzado por el comando "Empezar ADCS SW" que es idéntico al comando "Cambiar ADCS SW".

El *Beacon* se transmite cada 30 segundos de manera automática, haya o no operador escuchando. Por el contrario, la descarga de telemetría (TM) se hace siempre bajo petición de tierra. Además es importante destacar que un único comando de TM puede llevar aparejados *m* paquetes de TM a transmitir desde el QBITO a tierra. Tanto los comandos como las peticiones de TM se pueden enviar en cualquier momento, sin importar el punto dentro de la ejecución en la que se encuentra el SW del ADCS. Es por esto que, como ya se ha mencionado, la recepción de los telecomandos, así como la recepción y respuesta de las peticiones de telemetrías se tienen que hacer instantáneamente mediante una interrupción.

Figura 10.1. Diagrama de secuencia de funcionamiento del ADCS de QBITO.

Figura 10.2. Detalle del proceso de cambio de SW o arranque del ADCS.

En la figura 10.2 se detalla el proceso de arranque del ADCS, que aparece esquematizado en la parte inferior de la figura 10.1. El ADCS de QBITO tiene dos *bootloaders* (gestores de arranque), uno primario, que solo tiene interfaz UART, y otro secundario, que es arrancado automáticamente por el primario y que tiene interfaz I2C. Es este último *bootloader* con quien se comunica el OBC y el que se encarga de arrancar un SW u otro. Existen 7 slots de memoria flash donde se pueden guardar hasta 7 SW de control de actitud diferentes de hasta 512 KB. Para minimizar el riesgo de corrupción de SW por efecto de los eventos que afectan a un único bit (SEU por sus siglas en inglés: *Single Event Upset*), tanto el SW validado como el ESW se han duplicado en slots diferentes. Además, y como seguridad extra, se han subido también versiones diferentes de cada uno de ellos en un tercer slot: una versión más vieja del SW validado así como una versión del ESW con comunicación UART (en el ESW nominal se ha desactivado esta funcionalidad dado que a veces puede causar errores).

Con todo esto, el *bootloader* final del ADCS queda como se muestra en la figura 10.3, siendo el SW por defecto el ESW nominal sin UART, los espacios 1, 2 y 3 son los correspondientes al SW validado y los espacios del 4 al 6 para el ESW. El espacio 7 corrompía los binarios guardados en él por lo que se decidió no utilizarlo en órbita. Por último, en caso de que el SW del ADCS se reinicie sólo tres veces seguidas, la ejecución se interrumpe y se queda permanentemente en el segundo *bootloader* hasta que el OBC le comande salir.

	Index	Length	CRC Checksum	Description
▶	Internal Flash	448704	CC14	Final_NO_U...
	Ext Flash 1	230840	084E	SSC_2_1_2
	Ext Flash 2	230840	084E	SSC_2_1_2
	Ext Flash 3	230808	61FC	SSC_2_0
	Ext Flash 4	457544	40B6	Final_UART...
	Ext Flash 5	448704	CC14	Final_NO_U...
	Ext Flash 6	448704	CC14	Final_NO_U...

Figura 10.3. *Bootloader* del ADCS.

En la figura 10.2 se ha detallado el proceso de gestión del *Bootloader* del ADCS de QBITO. Este proceso se realiza tanto para arrancar el ADCS como para hacer un cambio de SW entre el SW validado o el ESW experimental. Este proceso se inicia siempre bajo recepción del correspondiente comando desde tierra. El primer paso consiste en comandar al EPS para que reinicie el módulo del ADCS de tal manera que salte automáticamente el primer *bootloader*. Siete segundos después de mandar el comando al EPS, el OBC inicia la comunicación con el segundo *bootloader* del ADCS. Una vez iniciada esa conversación el segundo *bootloader* aborta la ejecución del SW primario indefinidamente. En este momento, el OBC copia en la memoria flash interna el contenido del slot adecuado y reinicia el ADCS. En el siguiente arranque, tras ambos *bootloaders*, el SW elegido inicia su ejecución. Por último, el OBC se asegura que el SW cargado es el correcto, pone en hora el reloj interno del SW del ADCS y deja que se ejecute la secuencia nominal del SW del ADCS como se detalla más adelante.

Una vez acabada la secuencia de arranque del *bootloader*, empieza la ejecución del SW del ADCS propiamente dicho. En concreto nos vamos a centrar en el ESW por ser parte del núcleo de esta tesis. Su funcionamiento se esquematiza en la figura 10.4 donde aparecen como actores tanto módulos físicos (i. e. el OBC o módulos del ADCS como el CubeSense o el CubeControl) como funciones de SW dentro del ADCS. En la figura 10.4 aparecen los siguientes actores de izquierda a derecha:

- (i) OBC. En este esquema el OBC comanda al ADCS al recibir comando desde tierra (obviado en el esquema).
- (ii) ADCS *main*. Este módulo es la columna vertebral del ESW: hace de interfaz con el OBC y gestiona el resto de módulos del ADCS.
- (iii) ADCS coms. En este módulo se engloban todas las funciones de gestión de comandos y telemetrías.
- (iv) Func. ADCS. Aquí se engloban todas las funciones propias de la determinación y el control de actitud como se describen en las secciones siguientes.
- (v) Ciencia ADCS. En este módulo se engloba la gestión de los modos de operación (descritos en la tabla 10.1 y en la tabla 10.2) del ESW.
- (vi) Módulos ADCS. Este bloque representa a las placas CubeSense y CubeComputer.

Los primeros pasos tras el arranque consisten en inicializar todos los sistemas internos así como configurar las funciones de comunicaciones y las

funciones del ADCS propiamente dichas. Dentro de estas últimas funciones es donde están los algoritmos estudiados en esta tesis.

Figura 10.4. Esquema de secuencia de funcionamiento del ESW.

Código	Descripción
1	Mantener la actitud nominal (0, 0, 0) durante una órbita.
2	Establecer una actuación de diez grados en cabeceo: comando de (0, 10, 0). Espera quinientos segundos y vuelve a la actitud nominal.
3	Establecer una actuación de cincuenta grados en cabeceo: comando de (0, 50, 0). Espera quinientos segundos y vuelve a la actitud nominal.
4	Establecer una actuación de diez grados en balanceo: comando de (10, 0, 0). Espera quinientos segundos y vuelve a la actitud nominal.
5	Establecer una actuación de cincuenta grados en balanceo: comando de (50, 0, 0). Espera quinientos segundos y vuelve a la actitud nominal.
6	Establecer una actuación de diez grados en guiñada: comando de (0, 0, 10). Espera quinientos segundos y vuelve a la actitud nominal.
7	Establecer una actuación de cincuenta grados en guiñada: comando de (0, 0, 50). Espera quinientos segundos y vuelve a la actitud nominal.
8	Establecer una actuación de menos diez grados en los tres ángulos: comando de (-10, -10, -10). Espera quinientos segundos y vuelve a la actitud nominal.
9	Establece una actuación sinusoidal de amplitud veinte grados y periodo cinco mil segundos en cabeceo durante una órbita completa: $(0, 20 \sin\left(\frac{2\pi t}{5000}\right), 0)$.
10	Establece una actuación sinusoidal de amplitud veinte grados, valor medio de menos diez y periodo cinco mil segundos en cabeceo durante una órbita completa: $(0, 20 \sin\left(\frac{2\pi t}{5000}\right) - 10, 0)$.
11	Mantiene actitud nominal durante una órbita e intenta adquirir una imagen con la cámara de nadir, preferiblemente al sobrevolar el terminador.
12	N/A
13	Mantiene la actitud nominal durante H horas y, si durante ese periodo pasa encima de unas coordenadas geográficas modificables, toma una fotografía con la cámara de Nadir.

Tabla 10.1. Definición de las secuencias correspondientes a los test científicos.

Antes de continuar es importante remarcar que, a diferencia de las peticiones de telemetría que se responden inmediatamente, los comandos se

reciben y guardan en memoria pero se ejecutan solo en ventanas predefinidas dentro del bucle de control. La secuencia del bucle es siempre la misma:

- (i) Leer los datos de los sensores.
- (ii) Comandar los actuadores con la actuación calculada en el ciclo anterior.
- (iii) Procesar todos los comandos recibidos durante el ciclo anterior.
- (iv) Ejecutar todas las funciones de control del ADCS para calcular la nueva actuación.

El bucle de control se repite siempre de manera automática, por lo que si se quiere pausar su ejecución de manera temporal o permanente existen dos opciones, apagar el ADCS mediante comando al EPS o comandar un modo de *standby*. En este modo se sigue ejecutando todo el control cada segundo, pero se comandan los actuadores a cero para evitar que puedan interaccionar con el entorno.

La línea etiquetada como "Ciencia ADCS" de la figura 10.4 contiene la definición de los test científicos previstos para el experimento ESW (tabla 10.1) así como todos los modos de operación del ADCS en caso de que se usase como control de actitud para QBITO (tabla 10.2). Se entiende por test científico una secuencia determinada de comandos en actitud cuyo objetivo es estudiar el desempeño del algoritmo de control de actitud en órbita. Además, cada test graba en memoria una serie de variables temporales a frecuencias determinadas, determinadas para maximizar el retorno científico de la misión.

Además de los modos correspondientes a test científicos, existen otros modos de operación diseñados para la operación del SW como control de actitud nominal del satélite y que se encuentran detallados en la tabla 10.2.

Es interesante aclarar porqué para la toma de imágenes se usa un modo independiente. Dado que el bucle de control consume una fracción importante del tiempo disponible, no da tiempo a descargar la imagen desde el módulo CubeSense en un único ciclo. Por ello, para no alterar el funcionamiento del algoritmo de control ni el desempeño del satélite en general, solo una pequeña fracción de la imagen se transfiere en cada ciclo, repitiendo el proceso hasta haber completado la imagen (hasta cinco minutos para las fotos más grandes).

Código	Nombre	Descripción
0	Seguro o <i>standby</i>	Ejecuta todo el SW de control pero puede, o no, ejecutar los comandos calculados.
98	Nominal	Mantiene indefinidamente la actitud nominal.
99	B-dot	Ejecución mejorada del algoritmo B-dot para frenar al satélite.
100	Aceleración rueda	Acelera la rueda compensando el giro con los magnetopares.
101	Modo imán permanente	Establece una actuación fija en los magnetopares.
200	Tomar imagen	Mantiene actitud nominal y toma una foto.
201	Comprimir imagen	Comprime una imagen en N sub imágenes JPG.
254/255	Auto test	Se ejecuta un auto test.

Tabla 10.2. Modos de operación del ADCS de QBITO.

Como ya se ha visto, el SW se encarga de llamar a los módulos de control a intervalos fijos. A pesar de que se podría efectuar todo el control en un solo paso, se prefiere compartimentarlo en pequeños bloques que son llamados secuencialmente. Esto proporciona mayor control y flexibilidad en el proceso. Además existe una ventaja añadida que se apreció en etapas tardías del diseño y es el hecho de que permite usar variables de 64 bits solamente en los módulos imprescindibles, dejando el resto en 32 bits (con menor precisión), lo que permitió reducir considerablemente el tamaño del código autogenerado. Es importante remarcar que, aunque en general el paso de 64 a 32 bits siempre implica una pérdida de precisión, solo en muy pocos casos esta pérdida es significativa (i. e. únicamente si se usan variables con muchos dígitos significativos esta pérdida es importante). Por supuesto, en los casos en los que los errores son significativos se han seguido usando variables de doble precisión.

10.1.1. Funciones para el control de actitud: código Matlab autogenerado

El núcleo del SW del ADCS son cuatro módulos o bloques: propagador orbital, calculador de efemérides, determinación de actitud y control de actitud propiamente dicho (figura 10.5).

Figura 10.5. Esquema de "clases" de las funciones autogeneradas. Las flechas representan las dependencias entre bloques.

Sólo dentro de este último bloque se encuentra el algoritmo de control basado en lógica difusa estudiado en esta tesis. En la figura 10.5 se muestra el esquema de cada uno de estos bloques así como sus variables de entrada y de salida, junto con los flujos de datos que indican de donde vienen las variables de entrada y a donde van las de salida. En la figura 10.5 aparece un bloque que no se ha mencionado hasta ahora que es el filtro de Kalman (sección 3.3). Éste se engloba dentro del bloque de determinación, pero en realidad es un conjunto separado de funciones.

Para evitar tener que generar todas estas funciones, así como las funciones matemáticas asociadas en código C, se han desarrollado directamente en Matlab aprovechando las funciones ya implementadas para el DKE. Después se usa la funcionalidad dentro del entorno Matlab-Simulink que permite exportar los modelos generados con estas aplicaciones a código C. Por último, éste puede ser incorporado al SW del microprocesador con sólo pequeñas modificaciones. En las secciones siguientes se describen cada uno de los bloques principales que conforman las funciones de control de actitud de QBITO.

10.1.2. Propagador orbital

El primer código autogenerado es el propagador orbital. Aunque en el simulador se ha usado un modelo kepleriano con término J_2 , en el SW embebido se ha decidido utilizar un propagador más avanzado (SGP4 por sus siglas en inglés: *Simplified General Perturbations v4*) (Hoots, 1980) (Vallado, 2007). Esta decisión se toma porque el propagador SGP4 es un código ya validado por múltiples estudios y misiones, de manera que su uso reduce la posibilidad de errores en la misión. Además, el algoritmo SGP4 calcula de manera automática el decaimiento provocado por la resistencia atmosférica sin necesidad de módulos externos.

El propagador SGP4 toma como parámetros de entrada un juego de valores llamados TLE (por sus siglas en inglés: *Two Line Elements*). Estos son calculados diariamente por el Mando Norteamericano de Defensa Aerespacial (NORAD por sus siglas en inglés: *North American Aerospace Defense Command*) y codifican toda la información necesaria de la órbita para su propagación mediante el algoritmo SGP4. La información más detallada sobre cada uno de los parámetros del TLE se puede encontrar en la abundante bibliografía al respecto (Vallado, 2007) o en la página donde se publican diariamente (Celestrak, [online]). Como se acaba de indicar, estos

parámetros se pueden descargar de una página web, donde el NORAD los actualiza, aproximadamente cada día. De manera alternativa se pueden calcular a bordo usando lecturas del GPS. En la misión QBITO ambas posibilidades están contempladas, siendo la nominal la transmisión desde tierra del TLE actualizado. Sin embargo, cada vez que se activa el GPS el control estima los nuevos TLE de manera automática.

Sólo estos dos algoritmos, el propagador SGP4 y el que calcula los nuevos TLE usando lecturas del GPS, se han mantenido usando doble precisión dado que, por los parámetros que utilizan, era imprescindible si se deseaba usar los algoritmos sin modificaciones significativas.

El propagador SGP4 proporciona como salida las coordenadas en ejes ECI del satélite así como las coordenadas geográficas del mismo. Además, el módulo de propagación proporciona una serie de variables adicionales, no estrictamente relacionadas con la propagación SGP4. En concreto proporciona el cuaternión de rotación entre ejes ORF y ejes ECI así como la velocidad de rotación de uno respecto al otro.

10.1.3. Cálculo de efemérides

Una vez conocida la posición del satélite así como la fecha exacta, es fácil calcular las lecturas que deberían dar los distintos sensores. Esto se hace calculando mediante modelos la posición relativa del Sol respecto al satélite así como el vector de campo magnético instantáneo. Dado que la órbita es circular, el vector nadir siempre es $\mathbf{Z}_0$ por lo que es innecesario el uso de un modelo más avanzado.

El vector magnético se calcula utilizando el modelo IGRF 11 (Finlay, 2010) que, a pesar de estar obsoleto (el modelo actualizado está disponible en versiones más recientes de Matlab), sigue proporcionando estimaciones adecuadas para el control de actitud. El vector solar se calcula trivialmente en ejes ECI, por lo que únicamente resta convertirlo a ejes ORF. Además de estos modelos, este módulo estima la densidad de la atmósfera residual utilizando el modelo NRLMSISE-00 (Picone, 2002). La estimación de la densidad es necesaria para estimar los pares de perturbación debidos a la resistencia aerodinámica usados por el algoritmo DA. Sin embargo, cuando se estima la densidad atmosférica aún no se tiene conocimiento de la actitud del satélite, por lo que no se pueden estimar los pares de perturbación sufridos por el satélite necesarios para el control DA (sección 8.2). Por lo tanto, dichos

pares son calculados en el módulo de determinación que se explicará a continuación.

10.1.4. Determinación de actitud

Siguiendo la secuencia del control de la figura 10.5, una vez adquiridas las lecturas de los sensores, éstas se comparan con las estimaciones hechas por el anterior módulo para determinar la actitud. En el caso de QBITO, el módulo de determinación se ha separado en dos bloques, una determinación simple, seguida de un módulo de filtrado recursivo de tipo Kalman (Kalman, 1960). La razón principal para esta separación es que el filtro de Kalman conlleva una serie de algoritmos complejos que han tenido que ser desarrollados y que pueden contener errores. Por lo tanto, en caso de fallo en el funcionamiento del filtro de Kalman, este módulo se podría desactivar en órbita usando directamente la estimación de actitud no filtrada en el control difuso. Sin embargo, como en eclipse la gran mayoría de sensores no funcionan (i. e. solo el magnetómetro y el giróscopo siguen proporcionando lecturas), la única manera de disponer de estimaciones de actitud en eclipse es mediante algoritmos recursivos (i. e. filtro de Kalman). Por lo tanto si hubiese que desactivar dicho filtro, la funcionalidad del satélite quedaría limitada, pues en eclipse no habría determinación.

La entrada al bloque de determinación son las lecturas de todos los sensores del satélite, sus correspondientes matrices de transformación y calibración y, por último, los vectores obtenidos en el cálculo de efemérides previo. El módulo de determinación calcula de manera dinámica los pesos de cada lectura dependiendo del sensor en cuestión y sus márgenes operativos. Por lo tanto, si alguno de los sensores no proporciona lectura (v. g. el sol no está en el FOV correspondiente) basta con proporcionar un valor absurdo de la lectura y el algoritmo de determinación excluye automáticamente dicha lectura. Una vez normalizadas las lecturas y calculados los pesos de todas ellas se aplica el algoritmo Qmethod de Davenport (Keat, 1977) que, a diferencia del TRIAD, admite tantos pares de vectores lectura-modelo como se disponga, aumentando la precisión del algoritmo.

El algoritmo Qmethod es un estimador robusto que minimiza la función de pérdidas del problema de Wahba (Wahba, 1965). El problema de Wahba es la piedra angular de la determinación de actitud. Éste trata de encontrar una matriz ortogonal $\mathbf{A}$, con determinante positivo y unitario, que minimice la función de pérdidas (L) definida como en la ec. (10.1).

$$L(\mathbf{A}) = \frac{1}{2} \sum_i a_i |\mathbf{b}_i - \mathbf{A}\mathbf{r}_i|^2 \quad (10.1)$$

Donde el conjunto de las $\mathbf{b}_i$ y $\mathbf{r}_i$ son los vectores unitarios de la determinación, en ejes BRF (i. e. los que provienen de la lectura de los sensores) y en ejes de referencia (i. e. los que se han obtenido mediante modelos) respectivamente. Las ganancias a_i son los pesos de cada una de las medidas y suelen ir relacionados con la precisión de la misma.

La solución propuesta por Davenport (Keat, 1977) consiste en expresar la matriz $\mathbf{A}$ mediante el cuaternión correspondiente de la forma mostrada en la ec. (10.2). Si se calcula $\mathbf{B}$ como en la (10.4), se obtiene la ec. (10.3). Esto es interesante porque se demuestra que minimizar la traza que aparece en la ec. (10.3) es equivalente a minimizar la función de pérdidas de la ec. (10.1).

$$\mathbf{A} = (q_0^2 - |\vec{\mathbf{q}}|^2) \mathbf{I}_3 + 2\vec{\mathbf{q}}\vec{\mathbf{q}}^T - 2q_0[\vec{\mathbf{q}} \times] \quad (10.2)$$

$$\text{tr}(\mathbf{A}\mathbf{B}^T) = \mathbf{q}^T \mathbf{K} \mathbf{q} \quad (10.3)$$

$$\mathbf{B} = \sum_i a_i \mathbf{b}_i \mathbf{r}_i^T \quad (10.4)$$

La matriz $\mathbf{K}$ viene definida en la ec. (10.5), mientras que las correspondientes $\mathbf{S}$ y $\mathbf{z}$ están definidas a continuación en las ec. (10.6) y ec. (10.7).

$$\mathbf{K} \equiv \begin{bmatrix} \mathbf{S} - \mathbf{I} \text{tr}(\mathbf{B}) & \mathbf{z} \\ \mathbf{z}^T & \text{tr}(\mathbf{B}) \end{bmatrix} \quad (10.5)$$

$$\mathbf{S} \equiv \mathbf{B} + \mathbf{B}^T \quad (10.6)$$

$$\mathbf{z} = \sum_i a_i \mathbf{b}_i \times \mathbf{r}_i \quad (10.7)$$

Por último, el cuaternión que representa la actitud del satélite es el autovector normalizado de la matriz $\mathbf{K}$ correspondiente al máximo autovalor, lo que se puede resolver por múltiples métodos.

10.1.5. Filtro de Kalman

Como ya se ha mencionado, usando las medidas de actitud proporcionadas por el algoritmo de determinación de actitud, se diseña un

filtro de Kalman Extendido para mejorar la precisión de la estimación de la actitud. A grandes rasgos, el filtro de Kalman estima la actitud instantánea del satélite combinando cada nueva lectura con una predicción hecha al propagar lecturas pasadas. Esto quiere decir que sabiendo la actitud en el pasado, junto con las actuaciones que se han comandado, es posible propagar la actitud al instante actual. Como las fuentes de error son diferentes para las dos estimaciones de actitud (la actitud propagada y la actitud estimada previamente) al combinar ambas estimaciones se obtiene una tercera con una precisión mejor que las dos anteriores por separado. Esta tercera estimación es la que se llama actitud filtrada. Es fácil comprobar que, en caso de no disponer de medida de actitud (i. e. por estar en eclipse), el filtro de Kalman sigue proporcionando estimaciones al propagar actitudes anteriores.

Para poder filtrar satisfactoriamente la actitud estimada es necesario propagar las actitudes anteriores. Para ello es necesario conocer la velocidad angular del satélite. Sin embargo, en el subsistema de actitud solo se dispone de un giróscopo a un eje, alineado con el eje $\mathbf{Y}_B$. Por lo tanto, la velocidad angular debe deducirse de las estimaciones previas de la actitud por medio del algoritmo PSELIKA (Azor, 1998). Por último, usando la lectura del giróscopo en su eje, las estimaciones hechas por el PSELIKA y la actitud obtenida por el Qmethod, podemos hacer un filtro de Kalman que nos proporciona la actitud filtrada aun cuando no se dispone de lecturas de los sensores.

10.1.6. Control de actitud

Después del proceso de determinación (con o sin filtro de Kalman) se dispone de una estimación de la actitud del satélite y se puede ejecutar el algoritmo de control propiamente dicho. Éste es una versión ligeramente modificada del algoritmo que se ha estudiado en la sección 7.2. En concreto se ha añadido una función para calcular los comandos necesarios para los magnetopares a partir de la actuación solicitada en par. Por último, se ha añadido una opción para, en caso de avería de la rueda de momento, toda la actuación se lleve a cabo usando los magnetopares.

10.2. Implementación del SW embarcado

Una vez que se ha generado el código necesario para embeber en el microprocesador, éste debe ser integrado en el *hardware* de vuelo. En esta sección se va a hacer una pequeña introducción a dicho proceso.

10.2.1. Ensayos en placa de prototipado

El SW embarcado es capaz de controlar todos los periféricos, tanto del CubeComputer como del resto de módulos del ADCS. Es especialmente crítico el hecho de poder manipular la memoria flash o el *bootloader* dado que un error en esas zonas puede causar daños no recuperables en órbita. Por lo tanto, resulta imprescindible no ejecutar código dudoso en el módulo del ADCS directamente (así como en ningún otro módulo de QBITO). De esta manera, el código autogenerado, así como el resto de funciones de gestión, deben ser probadas antes de ser ejecutadas en el *hardware* de vuelo para garantizar que no se van a producir daños en el mismo. Por lo tanto antes de embeber el código éste debe ser ejecutado en una placa de prototipado (DVK por sus siglas en inglés: *Development Kit*) que cuenta un microprocesador muy parecido al del módulo Cubecomputer, un ARM Cortex-M3 de 32 bits. De esta manera, el SW que se prueba en el módulo de vuelo ya ha sido probado en un procesador equivalente, a modo de modelo de ingeniería. Además, el proceso de desarrollo se ve acelerado ya que la placa de prototipado puede ser usada en cualquier tipo de entornos (a diferencia del módulo del ADCS que debe permanecer en entornos controlados siempre que sea posible). Además, el DVK permite la depuración de código embarcado con lo que se avanza significativamente al poder usar esta función para encontrar fallos en el código. Por último, el DVK admite alimentación genérica externa, bien por USB o bien por toma de corriente a diferencia de la unidad CubeComputer que debe ser alimentada a 3.3V y 5V mediante fuente de alimentación regulada.

10.2.2. Implementación en el CubeComputer

La placa de CubeComputer es el cerebro del módulo del ADCS. Las principales características de éste son:

- Microcontrolador basado en un ARM Cortex-M3 de 32 bits.

- Frecuencia de hasta 48 MHz.
- Hasta 512 KB para el código embarcado.
- 2 MB de memoria volátil (SRAM por sus siglas en inglés: *Static Random Access Memory*) para ejecución de código.
- Dos buses I2C y un bus UART.

Lo primero que llama la atención es el poco espacio disponible para código embarcado. Sobre todo teniendo en cuenta que todo el código autogenerado funciona, por defecto, con doble precisión. Es por esta razón por la que se decide que la mayoría del código autogenerado funcione con precisión simple, dado que el resultado obtenido no difiere sensiblemente del que se obtendría al usar doble precisión. Sin embargo, en los dos módulos que se manejan magnitudes orbitales y tiempos es necesario mantener precisión doble. A modo de ejemplo, en precisión simple, a la altura de órbita de QBITO, el fondo de escala es de medio metro en distancias o de hasta 2 minutos en tiempo UNIX. Usando precisión doble solo donde es necesario se reduce el código lo suficiente como para poder ser embarcado en el espacio disponible. Además de esto, el código compilado puede ser optimizado en espacio por el compilador. Sin embargo, si se sobrepasa un cierto umbral de optimización, el código se vuelve errático y se puede llegar al caso (real) de que un fallo en el SW embarcado de actitud dejó QBITO congelado y sin posibilidad de recuperación sin intervención física.

Esto pone de manifiesto el hecho de que el sistema del ADCS debe ser lo suficientemente robusto, no solo para asegurar su buen, sino también para no afectar el resto del satélite. Debido a este error se implementó una función de vigilancia estricta, por la que si en 8 segundos no se ejecuta una determinada rutina, el SW se reinicia automáticamente.

Como ya se ha mencionado, el SW del ADCS, así como el resto de QBITO funcionan usando el tiempo UNIX. Éste mide los segundos (o milisegundos) empleando un origen de tiempo determinado, en concreto, la medianoche del uno de Enero de 1970. Este sistema permite llevar cuenta del tiempo usando un entero sencillo, sin preocuparse de cambios de mes, año o incluso días bisiestos. Dado que el ADCS no hace uso de memoria no volátil, salvo para los parámetros de configuración que se guardan en una tarjeta de memoria no volátil, con cada reinicio el calendario arranca en la misma fecha. Por lo tanto, el OBC debe establecer la hora correcta al arrancar el SW del ADCS, como se observa en la figura 10.2.

11. QBITO EN ÓRBITA

11.1. Introducción

En este capítulo se aborda la fase final del satélite QBITO desde el proceso de fabricación, hasta su operación en órbita, pasando por la integración en el módulo de despliegue, el lanzamiento y despliegue desde la ISS.

11.2. Proceso previo

En esta sección se va a recoger todo el proceso necesario antes del inicio de las operaciones en órbita propiamente dichas.

11.2.1. Fabricación e Integración

El primer paso antes de poder ser integrado en un lanzador es el montaje del satélite (figura 11.1 y figura 11.2). El montaje de QBITO se hace en dos fases, la primer se lleva a cabo en el E-USOC y consiste en el montaje e integración de todos los subsistemas y experimentos salvo el INMS. La segunda se lleva a cabo en ISIS y consiste en el montaje e integración del INMS y el *fit-check* en el dispensador. Finalmente, Nanoracks hace la integración con el lanzador (figura 11.3).

En la figura 11.1 se muestra al satélite QBITO sin paneles laterales. En el extremo izquierdo de la imagen se pueden apreciar parte de las antenas: el monopolo de VHF saliendo hacia la izquierda y dos de los cuatro monopolos de UHF saliendo hacia arriba y hacia el lector. Hacia la mitad del satélite se aprecia la cámara de nadir y, justo debajo, la rueda de momento. Por último, a la derecha de la imagen, pegado al panel lateral, se aprecia una de las dos probetas del experimento PCM, recubiertas por material reflectante y cinta Kapton por razones térmicas.

Figura 11.1. QBITO sin los paneles laterales.

En la figura 11.2 se observan las caras correspondientes a los ejes $-X_B$, $+Y_B$ y $+Z_B$. Es importante resaltar que en la figura 11.2 el satélite está "boca abajo" respecto a los ejes BRF que son los usados por el ADCS. El módulo del INMS se ensambla en la parte inferior de la fotografía. En la parte derecha de la imagen, la correspondiente el eje $+Y_B$, se aprecia la parte exterior de las dos probetas del experimento PCM. En la parte izquierda se tiene la cara correspondiente al eje $+Z_B$ donde se aprecian tanto la cámara de nadir del ADCS como el magnetómetro y su montura. Además, en esta cara también aparecen unas placas de latón, ya que por requisitos impuestos por el SW de control de actitud nominal (i. e. el proporcionado por SSC) fue necesario añadir una serie de contrapesos al satélite ya finalizado para que los momentos de inercia fuesen los exigidos. Además, como se observa en la figura 11.2 el magnetómetro tuvo que ser forrado de material reflectante para que su temperatura de funcionamiento estuviese dentro de los márgenes de operación.

Figura 11.2. QBITO ya finalizado, a falta de integrar con el INMS.

En la figura 11.3 se tiene un primer plano del experimento INMS integrado con QBITO. Es conveniente resaltar que en esta imagen los paneles solares se hayan cubiertos para evitar dañarlos o degradarlos. Sin embargo, hacia la mitad del satélite, en la parte inferior de la imagen se aprecian, a la izquierda los conectores de comunicaciones y a la derecha la cámara solar y la antena del GPS.

Figura 11.3. INMS integrado con QBITO.

El siguiente paso consistió en la integración de QBITO en el dispensador de NanoRacks. Éste es un contenedor cerrado y seguro para la fase de lanzamiento que tiene capacidad para alojar hasta doce CubeSats de 3U. Dicho dispensador sube a la ISS a bordo de la nave Cygnus, montada sobre el lanzador Atlas V (figura 11.4).

Figura 11.4. Despegue del lanzador Atlas V con la nave Cygnus en su interior el día 18 de Abril de 2.017. Crédito: *United Launch Alliance*.

Una vez a bordo de la ISS, el brazo robótico japonés sitúa el dispensador en el exterior de la estación, e inyecta los satélites en órbita. La mayoría de los satélites de la misión QB50 son desplegados de esta forma, en dos tandas separadas alrededor de dos semanas.

Figura 11.5. Despliegue de QBITO (abajo) desde el dispensador de Nanoracks (arriba).

11.2.2. Lanzamiento y despliegue

El día 18 de Abril de 2017 la nave Cygnus 7, propiedad de *Orbital ATK*, despegó de Cabo Cañaveral a bordo del lanzador Atlas V, propiedad de la *United Launch Alliance*. Cuatro días después, el 22 de Abril, la nave Cygnus fue capturada por la ISS.

El día 24 de Abril el dispensador de Nanoracks, con QBITO dentro, fue instalado en el brazo robótico e instalado en el exterior del laboratorio JEM japonés de la ISS.

Por último, a las 11:55 GMT (hora media de Greenwich, por sus siglas en inglés: *Greenwich Mean Time*) del 25 de Abril QBITO fue inyectado en órbita (figura 11.5) y se iniciaron sus operaciones espaciales.

La maniobra de despliegue inyectó a QBITO en una órbita similar a la seguida por la ISS, proporcionándole un impulso de aproximadamente 2 m/s en una dirección que forma 45° con la tangente de la órbita, hacia “atrás” y hacia “abajo” (i. e. según el vector $\begin{bmatrix} 1 & 0 & 1 \end{bmatrix}$ del sistema ORF).

11.3. Operaciones en órbita y análisis de resultados

Debido a los retrasos ocurridos en la misión QB50 no ha sido posible incluir los resultados de los test en órbita. Se espera que éstos sean realizados durante la segunda mitad del año 2017.

Como ya se ha mencionado en la tabla 10.1, en el SW embarcado existen una serie de secuencias destinadas al ensayo y verificación tanto del SW como del algoritmo difuso. Los ensayos que se realizarán se agrupan en tres categorías:

- (i) Comisionado y estabilización.
- (ii) Operaciones nominales.
- (iii) Test científicos.

El primero de ellos, aunque ya habrá sido realizado previamente por el SW de SSC, consiste en verificar que todos los subsistemas (i. e. sensores, actuadores y módulos del ADCS) funcionan correctamente. A continuación se ejecutan una serie de algoritmos diseñados para frenar el satélite, orientarlo en la actitud nominal y acelerar la rueda de momento. Todos estos pasos son imprescindibles para una correcta operación del SW nominal.

El siguiente ensayo consiste en evaluar si el SW embarcado es capaz de ser usado como control de actitud nominal. Esto quiere decir que es capaz de mantener la actitud exigida por el INMS y ejecutar el resto de funciones de soporte necesarias (v. g. gestión de datos en la SD). Esto es importante porque todas las secuencias científicas parten y acaban en ese modo de control nominal.

Por último, una vez comprobado que el SW funciona correctamente, se ejecutarán las secuencias científicas todas las veces que sea posible para aumentar la confianza en los resultados obtenidos. Los datos científicos se

almacenan en una memoria flash no volátil y deben ser descargados a tierra. Sin embargo, dado el escaso ancho de banda disponible, se ha añadido a los paquetes de telemetría del ADCS información suficiente para hacer un análisis del desempeño del algoritmo. Una vez que todos los datos estén en tierra deberán ser procesados para validar tanto el DKE como el algoritmo desarrollado.

12. CONCLUSIONES

El conseguir un algoritmo de control fiable y eficiente suele ser una prioridad en prácticamente cualquier aplicación de un sistema de control. En el caso del control de actitud de un satélite, este hecho es todavía más relevante dado que la misión completa puede verse comprometida por un control defectuoso. Dado que las misiones espaciales tienen costes, en muchos casos astronómicos, maximizar el retorno de un satélite se convierte en un objetivo prioritario.

Por ello, en una abrumadora mayoría de casos, se suele sacrificar una mejora hipotética en eficiencia en aras de la confianza que da el uso de algoritmos ya probados. Esto dificulta la experimentación de nuevos algoritmos que puedan resultar más eficientes y que, por tanto, aumenten el retorno de la misión.

En esta tesis se ha estudiado el uso de algoritmos inteligentes, basados en lógica difusa, para el control de actitud de un nanosatélite en órbita LEO. Estos algoritmos son de limitada aplicación a día de hoy en satélites en órbita y con esta tesis se intenta profundizar en su conocimiento para facilitar futuras implementaciones del mismo. En este estudio se compara su eficiencia y precisión con la de un control de actitud clásico, un PID, ampliamente extendido en la industria, incluida la espacial. El objetivo final de este estudio es el poder implementar y probar un control basado en lógica difusa en una misión real, la misión del satélite QBITO, parte del proyecto QB50. Se ha escogido esta misión dado que, al ser un proyecto de presupuesto reducido, permite mayor capacidad de experimentación que grandes proyectos con presupuestos multimillonarios.

Para la consecución de los objetivos de esta tesis, en un primer paso se ha desarrollado un simulador de la dinámica de actitud de un nanosatélite sometido a perturbaciones ambientales. Sobre ese simulador se han diseñado y calibrado los dos algoritmos de control que se presentan en esta tesis: un algoritmo de control basado en lógica difusa y un algoritmo de control clásico de tipo PID. Ambos algoritmos de control se han estudiado en una serie de escenarios predefinidos. Los resultados obtenidos muestran que el PID es significativamente menos eficiente (en términos de función de coste) que el control basado en lógica difusa propuesto (se necesitan hasta tres veces más de energía al usar el PID según el escenario estudiado).

Sin embargo, el proceso de calibración utilizado ha sido demasiado activo optimizando el control difuso, llegando a afectar a su precisión, sobre todo a órbitas bajas donde los pares de perturbación crecen de manera exponencial. Para conseguir un control que cumpliera con los requisitos de la misión estudiada en todo el rango de alturas, se ha modificado el control difuso. El nuevo algoritmo inteligente, llamado Difuso-Adaptativo, adapta sus ganancias de manera dinámica según estima las perturbaciones ambientales. De esta manera, el control DA es capaz de mantener el requisito de precisión durante toda la misión, a un coste incluso menor que el algoritmo difuso original.

Seguidamente se ha comparado este nuevo algoritmo adaptativo frente al control PID original. Los resultados indican que este algoritmo es tan preciso o más que el PID, siendo significativamente más eficiente. En concreto el algoritmo DA es capaz de mantener el error residual por debajo de 0.01 rad consumiendo hasta un 70% menos de energía en maniobras sencillas que el PID.

A continuación se han empleado herramientas del análisis estadístico aleatorio para profundizar en la comparación de ambos controles, el difuso y el PID en diferentes escenarios. El estudio ha consistido en analizar la respuesta de ambos controles frente a una serie de incertidumbres en el diseño y el proceso de fabricación y montaje así como posibles escenarios de la misión. Para este análisis se ha utilizado el método Monte Carlo. Los resultados indican que el control difuso es mejor que el PID en un amplio rango de los parámetros. En términos medios, el PID es hasta un 50% menos eficiente y un 250% menos preciso que el difuso. Además, analizando los resultados de ambos algoritmos de control para cada una de las simulaciones, se observa que en un 65.2% de los escenarios posibles el control difuso es más preciso y más eficiente que el PID. La situación opuesta se da solo en un 2.9% de los escenarios estudiados.

Los resultados obtenidos hasta este punto indican que el control difuso propuesto puede ser un firme candidato para su uso en órbita como control de actitud. Estos resultados deben ser refrendados por ensayos en órbita. Sin embargo, debido a retrasos en la misión de QBITO ha sido imposible su estudio en órbita hasta la fecha.

Para poder probar el algoritmo difuso en órbita es necesario desarrollar un SW embarcado a bordo del QBITO. Éste se compone de dos bloques conceptualmente separados: funciones autogeneradas hechas en el entorno de Matlab-Simulink y otras funciones hechas directamente en C. Éstas últimas

son todas aquellas funciones de interacción con el *hardware* del ADCS (v. g. comunicaciones I2C con el módulo del OBC). Por otro lado, las funciones autogeneradas contienen todo lo relacionado con el control de actitud del satélite, incluido el algoritmo DA estudiado en esta tesis.

Por lo tanto, queda como trabajo futuro el estudio y análisis del control difuso en órbita. Éste se espera que pueda ser llevado a cabo en la segunda mitad del año 2.017.

Además, se propone como continuación directa del estudio teórico realizado en esta tesis el uso de algoritmos genéticos en el proceso de calibración de ambos controles para conseguir múltiples puntos de operación óptimos. De esta manera, la calibración podría ser modificada de manera dinámica en función de los requisitos instantáneos de la misión. También se propone el uso de algoritmos genéticos para la optimización de las funciones de pertenencia que definen el control-

Como trabajos futuros a más largo plazo están el extender el control difuso no solo al control de actitud, sino también al control de órbita, y comparar su eficiencia con la de otros controladores empleados actualmente.

13. BIBLIOGRAFÍA

Armendáriz, I. (2015). *Análisis paramétrico determinista y aleatorio de problemas dinámicos en estructuras aeroespaciales (tesis doctoral)*. Universidad Politécnica de Madrid.

Äström, K.J. y Hägglund, T. (1995). *PID Controllers: theory, Design, and Tuning 2ª ed.* ISA, ISBN 978-1-556-17516-9.

Azor, R. et al. (1998). *Angular-Rate Estimation Using Quaternion Measurements*. NASA technical documents.

Barrios, I., Laverón, A., Moreno, E. (2013). *QBITO Critical Design Review*.

Barrios, I., Laverón, A., Moreno, E. (2014). *QBITO, the first CubeSat by Universidad Politécnica*. Small Satellites and Services Symposium.

Beals, G.A. et al. (1988). *Hubble Space Telescope precision pointing control system*. Journal of Guidance, Control, and Dynamics, Vol. 11, No. 2, pp. 119-123.

Belk, C.A. et al. (1997). *Meteoroids and Orbital Debris: effects on Spacecraft*. NASA Reference Publication 1408.

Black, H.D. (1964). *A passive system for determining the attitude of a satellite*. AIAA Journal, Vol. 2, No. 7, pp. 1350-1351.

Celestrak. *Two Line Elements*. Disponible en: <http://celestrak.com>.

Cheng, C., Shu, S. y Cheng, P. (2009). *Attitude control of a satellite using fuzzy controllers*. Expert Systems with Applications, Vol. 36, No. 3 (2), pp. 6613–6620.

Crespo, J. (2011). *Diseño de controladores de sistemas multivariables basados en técnicas de asignación de autoestructuras aplicados a vehículos espaciales rígido-flexibles (tesis Doctoral)*. Escuela Técnica Superior de Ingeniería Informática UNED.

CubeSat, Program. *The CubeSat Standard*. Disponible en: <http://www.CubeSat.org/>.

Danchik, R.K. (1998). *An Overview of Transit Development*. Johns Hopkins APL Technical Digest, Vol. 19, No. 1, pp. 18-26.

E-USOC. *The Spanish User Support and Operations Centre*. Disponible en: <http://www.eusoc.upm.es>.

Fazlyab, A.R., Saberi, F.F. y Kabganian, M. (2016). *Adaptive attitude controller for a satellite based on neural network in the presence of unknown external disturbances and actuator faults*. Advances in Space Research, Vol. 57, No. 1, pp. 367-377.

Finlay, C.C. *et al.* (2010). *International Geomagnetic Reference Field: the eleventh generation*. Geophysical Journal International, Vol. 183, No. 3, pp. 1216-1230.

Gilmore, D.G. (2002). *Spacecraft Thermal Control Handbook, Volume I: fundamental Technologies*. AIAA, ISBN 978-1-884989-11-7.

Guana, P., Liub, X.J. y Liub, J.Z. (2005) *Adaptive fuzzy sliding mode control for flexible satellite*. Engineering Applications of Artificial Intelligence, Vol. 18, pp. 451-459.

Harland, D.M. y Lorenz, R. (2005). *Space Systems Failures: disasters and Rescues of Satellites, Rocket and Space Probes*. Praxis, ISBN 978-0-387-27961-9.

Hoots, F.R. y Roehrich, R.L. (1980). *Spacetrack Report #3: models for Propagation of NORAD Element Sets*. Aerospace Defense Command Peterson AFB CO Office of Astrodynamics.

Jantzen, J. (2013). *Foundations of Fuzzy Control: a Practical Approach 3^o ed.* Wiley, ISBN: 978-1-118-50622-6.

JAXA (2016). *Operation Plan of X-ray Astronomy Satellite ASTRO-H (Hitomi)*. Disponible en: http://global.jaxa.jp/press/2016/04/20160428_hitomi.html.

JCGM (2008). *Evaluation of measurement data — Guide to the expression of uncertainty in measurement*. Bureau International des Poids et Mesures, Vol 101.

Kalman, R.E. (1960). *A New Approach to Linear Filtering and Prediction Problems*. Transactions of the ASME - Journal of Basic Engineering, Vol. 82, pp. 35-45.

Keat J.E. (1977). *Analysis of least-squares attitude determination routine DOAOP*. Technical Report CSC/TM-77/6034, Computer Sciences Corp.

Layne, J. y Passino, K. (1993). *Fuzzy Model Reference Learning Control for Cargo Ship Steering*. IEEE Control System, Vol. 13, No. 6, pp. 23-34.

Lee, C.C. (1990). *Fuzzy Logic in Control Systems: fuzzy Logic Controller-Part I*. IEEE Transactions on Systems, Man, and Cybernetics, Vol. 20, pp. 404-418.

Lovera, M. (2015). *Magnetic satellite detumbling: the b-dot algorithm revisited*. 2015 American Control Conference. DOI: 10.1109/ACC.2015.7171005.

Mamdani, E.H. y Assilian, S. (1975). *An experiment in linguistic synthesis with fuzzy logic controller*. International Journal of Man-Machine Studies, Vol. 7, No. 1, pp-1-13.

Metropolis, N. y Ulam, S. (1949). *The Monte Carlo Method*. Journal of the American Statistical Association, Vol. 44, No. 247, pp. 335-341.

Nagi, F. *et al.* (2010). *Fuzzy bang-bang relay controller for satellite attitude control system*. Fuzzy Sets and Systems, Vol. 161, pp. 2104-2125.

Olsson, D. (1993). *A power system for a microsatellite*. Small Satellites Systems and Services, pp. 557-566.

Ortega, G. (1995). *Fuzzy logic techniques for rendezvous and docking of two geostationary satellites*. Telematics and Informatics, Vol. 12, No. 3(4), pp. 213-227.

Oshima, H., Yasunobu, S. y Sekino, S. (1988). *Automatic train operation system based on predictive fuzzy control*. IEEE International Workshop on Artificial Intelligence for Industrial Applications, pp. 485-489, DOI: 10.1109/AIIA.1988.13336.

Østergaard, J. (1976). *Fuzzy Logic Control of a Heat Exchanger Process*. Fuzzy Automata and Decision Processes, Amsterdam: north-Holland, pp. 285-320.

Picone, J.M., Hedin, A.E. y Drob, D.P. (2002). *NRLMSISE-00 empirical model of the atmosphere: statistical comparisons and scientific issues*. Journal of Geophysical Research, Vol. 107.

Robertson, R.E. (1958). *Gravitational Torque on a Satellite Vehicle*.

Ross, T.J. (2010). *Fuzzy Logic With engineering Applications*. John Wiley & Sons, Ltd. ISBN: 978-0-470-74376-8.

Shuster, M.D. (1978). *Approximate algorithms for fast optimal attitude computation*. Guidance and Control Conference, disponible en: <http://dx.doi.org/10.2514/6.1978-1249>.

Shuster, M.D. y Oh, S.D. (1981). *Three-axis attitude determination from vector observations*. Journal of Guidance, Control, and Dynamics, Vol. 4, No. 1, pp. 70-77.

Sidi, M.J. (2000). *Spacecraft Dynamics and Control: a practical Engineering Approach*. Cambridge University Press, ISBN-13: 978-0-521-78780-2.

Souza, L.C.G. (2006). *Design of Satellite Control System Using Optimal Nonlinear Theory*. Mechanics Based Design of Structures and Machines, Vol. 34, No. 4, pp. 351-364.

Stacey, F.D. y Davis, P.M. (2008). *Physics of the Earth 4^o ed.* Cambridge University Press, ISBN: 978-0-521-87362-8.

Steyn, W.H. (1994). *Comparison of Low-Earth-Orbit Satellite Attitude Controllers Submitted to Controllability Constraints*. Journal of Guidance, Control, and Dynamics, Vol. 17, No. 4, pp. 795-804.

Takagi, T. y Sugeno, M. (1985). *Fuzzy Identification of Systems and Its Applications to Modeling and Control*. IEEE Transactions on Systems, Man, and Cybernetics, Vol. 1, pp. 116-132.

Tanaka, K. y Sugeno, M. (1992). *Stability analysis and design of fuzzy control systems*. Fuzzy Sets and Systems, Vol. 45, pp. 135-156.

Tribble, A.C. (2003). *The Space Environment: implications for Spacecraft Design*. Princeton University Press, ISBN: 978-0-691-10299-3.

Toshikazu, T. y Toshiharu, H. (1991). *A Practical Application of Fuzzy Control for an Air-Conditioning System*. International Journal of Approximate Reasoning, Vol. 5, pp. 331-348.

UML. *Unified Modeling Language*. Disponible en: <http://www.uml.org/>.

Vallado, D.A. (2007). *Fundamentals of Astrodynamics and Applications*. Space Technology Library vol. 21, ISBN 978-0-387-71831-6.

Van Allen, J.A. (1959). *Radiation Belts around the Earth*. Scientific American, Vol. 200, No. 3.

Von Karman Institute. *The QB50 mission*, Disponible en: <https://www.qb50.eu/>.

Wahba, G. (1965). *A Least-Squares Estimate of Satellite Attitude, Problem*. SIAM Review Vol. 7, No. 3, p. 409.

Walker, A.R., Putman, P.T. y Cohen, K. (2015). *Solely Magnetic Genetic/Fuzzy-Attitude-Control Algorithm for a CubeSat*. Journal of Spacecraft and Rockets, Vol. 52, No. 6, pp. 1627-1639.

Wertz, J.R. y Larson, W.J. (1992). *Space Mission Analysis and Design* 3^o ed. Space Technology Library, Vol. 2, ISBN-13: 978-1-881-88310-4.

Wingrove, R.C. (1963). *A study of guidance to reference trajectories for lifting reentry at supercircular velocity*. NASA Technical Report 151.

Wolfgram, D.F. y Pipher, M.D. (1980). *EPS analysis of nominal STS-1 flight*. NASA Technical Report 81001.

Zadeh, L.A. (1965). *Fuzzy Sets*. Information and Control, Vol. 8. pp. 338-353.

Zadeh, L.A. (1973). *Outline of a new approach to de analysis of complex system*. IEEE Transaction on System Man and Cybernetics, Vol. 3, No. 1, pp. 28-44.

Zadeh, L.A. (1975). *The concept of a linguistic variable and its application to approximate reasoning; part I, II, III*. Information Sciences, Vol. 8, No. 3, pp. 199-249; Vol. 8, No. 4, pp. 301-357; Vol. 9, No. 1, pp. 43-80.

Ziegler, J.G. y Nichols, N.B. (1993). *Optimum settings for automatic controllers*. Journal of Dynamics Systems, Measurement and Control, Vol. 115, No. 2 (B), pp.759-765.

Zou, A., Dev Kumar, K. y Hou, Z. (2010). *Quaternion-Based Adaptive Output Feedback Attitude Control of Spacecraft Using Chebyshev Neural Networks*. IEE Transactions on Neural Networks, Vol. 21, No. 9.

ANEXO A: SISTEMAS DE REFERENCIA

Un sistema de referencia es un juego de tres vectores unitarios mutuamente ortogonales. La principal clasificación que se puede hacer sobre ellos es la separación entre inerciales o no inerciales. El principal sistema de referencia inercial que se va a usar es el inercial centrado en la Tierra (ECI por sus siglas en inglés: *Earth Centered Inertial*). Asociado a éste, pero no inercial, tenemos el centrado y fijo a la Tierra (ECEF por sus siglas en inglés: *Earth Centered-Earth Fixed*) que se diferencia del ECI principalmente en la rotación terrestre. Por último, podemos definir una serie de ejes fijados al centro de masas del satélite, ya sea fijos a la rotación del cuerpo (ejes cuerpo o BRF por sus siglas en inglés: *Body Reference Frame*) o independientes de la misma (ejes órbita u ORF por sus siglas en inglés: *Orbital Reference Frame*).

A.1 Ejes inerciales o ECI

Éste es el principal sistema de referencia inercial. También se conoce como sistema de coordenadas celestiales (Figura A.1).

Figura A.1. Representación de los ejes ECI y ECEF durante el equinoccio de primavera.

El eje X de este sistema de referencia apunta hacia el equinoccio vernal o punto Aries (γ). El punto Aries es la proyección en el infinito de la recta que une el centro de la tierra y el centro del sol el equinoccio Vernal (i. e. equinoccio de primavera). La intersección entre el plano orbital (plano de la

Eclíptica) y el plano ecuatorial coincide con esta línea. El eje Z es el normal al plano de la Eclíptica en el sentido positivo del momento angular orbital. Por último, el eje Y completa un triedro a derechas.

A.2 Ejes fijos a la tierra o ECEF

Los ejes ECEF también están fijos al centro de la Tierra, pero en lugar de estar fijos a puntos celestiales inmóviles, éstos giran solidarios a la rotación terrestre.

El eje X apunta hacia las coordenadas de Latitud 0° y Longitud 0° , o lo que es lo mismo, a la intersección del meridiano de Greenwich y el ecuador. El eje Z apunta al polo norte geográfico (no confundir con el polo norte magnético). Por último, el eje Y completa el triedro a derechas. Los ejes ECEF están representados en la Figura A.1 junto con los ECI de los que se diferencian por la inclinación del ecuador respecto a la eclíptica y la rotación terrestre respecto a su eje de rotación (i. e. un giro fijo alrededor del eje X en ejes ECI y un giro variable con el tiempo alrededor del eje Z en ejes ECEF).

A.3 Ejes órbita u ORF

Los principales ejes fijos al satélite que se consideran en este estudio son los ejes órbita y los ejes cuerpo, ambos representados en la Figura A.2. En esta tesis a cada uno de los tres ejes que componen este sistema de referencia tiene nombre propio ($\mathbf{X}_O$, $\mathbf{Y}_O$ y $\mathbf{Z}_O$).

Los ejes órbita tienen su origen en el centro de masas del satélite, su eje $\mathbf{X}_O$ está orientado según la tangente a la órbita en cada instante y apuntando en el sentido de la velocidad orbital. En caso de que la órbita sea circular el eje $\mathbf{X}_O$ coincide con la dirección del vector velocidad ($\mathbf{V}$) si se desprecia el efecto de la resistencia aerodinámica. El eje $\mathbf{Y}_O$ es normal al plano orbital y apunta en el sentido opuesto al del momento angular orbital del satélite. Por último el eje $\mathbf{Z}_O$ apunta hacia el nadir completando el triedro a derechas.

Figura A.2. Representación de los ejes ECEF, ORF y BRF.

A.4 Ejes cuerpo o BRF

Los ejes cuerpo se suelen definir teniendo en cuenta la geometría del satélite así como la misión a desarrollar. Además pueden existir múltiples sistemas de ejes cuerpo (v. g. los ejes principales para la misión QBITO son iguales a los marcados por el estándar CubeSat y los utilizados en el ADCS son los aquí expuestos). QBITO es un satélite alargado en un eje pero prácticamente simétrico en los otros dos. La misión principal de QBITO consiste en la operación del INMS que se encuentra en uno de los extremos de ese eje alargado y que debe apuntar lo más paralelo posible al vector velocidad orbital, siendo el giro sobre ese eje alargado indeterminado. Además, dado que el principal actuador es una rueda de momento, lo lógico es que el eje de dicha rueda sea paralelo al eje $\mathbf{Y}_O$ puesto que es el único eje constante (respecto al sistema ECI) a lo largo de la órbita. Igual que con los ejes ORF, cada uno de los ejes del sistema BRF tiene nombre propio ($\mathbf{X}_B$, $\mathbf{Y}_B$ y $\mathbf{Z}_B$).

Por lo tanto, los ejes cuerpo se definen fijos al centro de masas del satélite, con el eje $\mathbf{Y}_B$ alineado con el eje de rotación de la rueda de momento, el eje $\mathbf{X}_B$ alineado con el eje del INMS y el eje $\mathbf{Z}_B$ formando un triedro a derechas. De esta forma, se podría pensar en los ejes cuerpo como en los

típicamente usados en aviación, siendo el $\mathbf{X_B}$ hacia adelante, el $\mathbf{Y_B}$ hacia la derecha y el $\mathbf{Z_B}$ hacia abajo. Por lo tanto, en condiciones de actitud nominal, ORF y BRF será coincidentes.